Demosaicking Method for Multispectral Images Based on Spatial Gradient and Inter-channel Correlation

Shu Ogawa[1(✉)], Kazuma Shinoda[1], Madoka Hasegawa[1], Shigeo Kato[1], Masahiro Ishikawa[2], Hideki Komagata[2], and Naoki Kobayashi[2]

[1] Graduate School of Engineering, Utsunomiya University, 7-1-2 Yoto, Utsunomiya, Tochigi 321-8585, Japan
l4ogawa@mclaren.is.utsunomiya-u.ac.jp
[2] Faculty of Health and Medical Care, Saitama Medical University, 1397-1 Yamane, Hidaka, Saitama 350-1241, Japan

Abstract. Multispectral images have been studied in various fields such as remote sensing and sugar content prediction in fruits. One of the systems that captures multispectral images uses a multispectral filter array based on a color filter array. In this system, demosaicking processing is required because the captured multispectral images are mosaicked. However, demosaicking is more difficult for multispectral images than for RGB images owing to the low density between the observed pixels in multispectral images. Therefore, we propose a demosaicking method for multispectral images based on spatial gradient and inter-channel correlation. Experimental results demonstrate that our proposed method outperforms the existing methods and is effective.

Keywords: Demosaicking · Multispectral filter array · Interpolation · Inter-channel correlation · Spatial gradient

1 Introduction

Multispectral images (MSIs) consist of a higher number of color components than RGB images. Therefore, MSIs have been studied in fields that require accurate color representation such as remote sensing and the diagnosis by pathological images [1].

Various systems have been proposed for the capture of MSIs. These systems can be classified into the following three categories: (i) multi-camera-one-shot systems, (ii) single-camera-multi-shot systems, and (iii) single-camera-one-shot systems [2]. In this paper, we focus on the single-camera-one-shot systems that obtain an MSI by using a single image sensor with a multispectral filter array (MSFA). An MSFA is based on the color filter array (CFA) used for capturing RGB images. Therefore, mosaicked MSIs that have a single band at each pixel location can be captured by using an MSFA. The MSI can be obtained faster with this system than with the single-camera-multi-shot system. Further, the cost of capturing the MSI is lower than in the case of multi-camera-one-shot systems that use multiple cameras. Demosaicking processing is required in order to

A. Mansouri et al. (Eds.): ICISP 2016, LNCS 9680, pp. 157–166, 2016.
DOI: 10.1007/978-3-319-33618-3_17

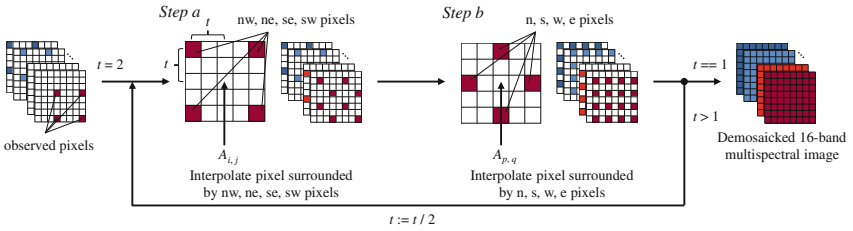

Fig. 1. Interpolation flow of Step 1 in MLDI.

restore the missing pixels. However, demosaicking is more difficult for MSIs than for RGB images because the density between observed pixels for the MSIs captured by an MSFA is lower than that for the RGB images. Earlier studies [3–5] can be applied to any MSFA, however, these methods are based on either inter-channel correlation [3, 4] or spatial gradient [5]. Although the methods based on both have been proposed [6, 7], it is difficult to apply them to any MSFA because they are demosaicking methods proposed for a particular MSFA.

In this paper, we propose multispectral local directional interpolation (MLDI) for some MSFAs. This method is based on local directional interpolation (LDI) that considers spatial gradient and inter-channel correlation, which is proposed by Zhang et al. [8] for CFAs. Experiments performed on test images by using our proposed method and existing methods demonstrate the effectiveness of the proposed method.

The rest of the paper is organized as follows: In Sect. 2, we describe the proposed demosaicking method. We present and discuss the experimental results in Sect. 3, and state the conclusion in Sect. 4.

2 Multispectral Local Directional Interpolation (MLDI)

2.1 Overview of Proposed Demosaicking Method

In this section, we explain the demosaicking algorithm of MLDI that consists of three steps: (1) First, the missing pixels are interpolated by extending LDI; (2) Next, the restored MSI obtained in Step 1 is updated by using the eight neighboring pixels; (3) Then, the artifacts of the restored MSI obtained in Step 2 are removed by using a median filter.

In Step 1, the missing pixels of the mosaicked MSI are interpolated by extending LDI. In this step, we assume that a $2^x \times 2^x$ (x = 1, 2, …)-band MSFA is arranged in $2^x \times 2^x$ pixels to interpolate the canter pixel surrounded by the four observed or previously interpolated pixels. Figure 1 shows an overview of Step 1. First, the center pixel surrounded by the peripheral pixels named *north-west* (*nw*), *north-east* (*ne*), *south-east* (*se*), and *south-west* (*sw*) pixels is interpolated. Then, the center pixel surrounded by the peripheral pixels named *north* (*n*), *south* (*s*), *west* (*w*), and *east* (*e*) pixels is interpolated. This process is repeated until all the missing pixels are

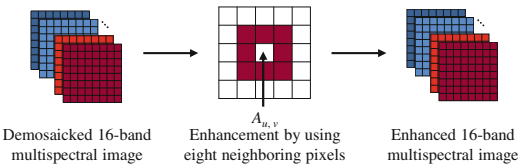

Fig. 2. Updating the restored image of Step 1 by using eight neighboring pixels (Step 2 in MLDI).

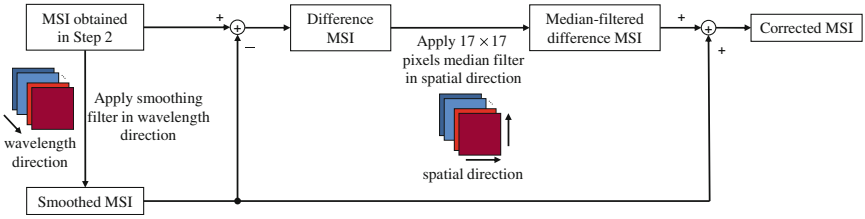

Fig. 3. Removing artifacts in the image obtained in Step 2 (Step 3 in MLDI).

interpolated because the density between observed pixels of the mosaicked MSI is lower than that of the mosaicked RGB image. The interpolation in Step 1 is based on the interpolation of red and blue components in LDI, which use previously interpolated pixels in green component. However, MSFA do not necessarily contain a band that the density between observed pixels is high such as the green component. Therefore, in Step 1, we use a band observed in the interpolation target pixel instead of green component in order to use the observed values at the same pixels as the pixels in green component used in LDI. However, this band may not be observed in pixels used for the interpolation because all pixels in green component have been already interpolated. Therefore, this band is interpolated previously when it is used.

Then, in Step 2, we improve the quality of the restored image obtained in Step 1 by using the eight neighboring pixels. Figure 2 shows an overview of Step 2. In Step 1, the bands and pixels used for each interpolation are different because the interpolation target band, the band observed in the interpolation target pixel and previously interpolated pixels are mainly used for other interpolation. Therefore, in Step 2, we use more bands and pixels than that in Step 1 by using the eight neighboring pixels that have been already interpolated in Step 1.

Finally, in Step 3, we apply a median filter to correct impulse noise as shown in Fig. 3, which is based on the chromatic regularity step of Buades et al. [9]. The chromatic regularity step of [9] decomposes RGB into YUV components, applies a 3×3 median filter to UV components, then recomposes the filtered UV into RGB components. Instead of UV components, we use the difference between the input and smoothed MSI, and remove the noise by using a median filter.

In Sects. 2.2, 2.3, and 2.4, we provide a detailed description of Step 1, Step 2, and Step 3, respectively.

2.2 Interpolation of Missing Pixels

Let the band targeted for interpolation be A, the position of the pixel targeted for interpolation be (i, j) as shown in Fig. 1 (*Step a*), and the reference band be S. Here, the reference band denotes the band in which the pixel value was observed at (i, j) in the mosaicked MSI. In Step 1, we predict the difference between the pixel values of the target and reference bands at (i, j), and calculate the target pixel value by using the difference value at (i, j). First, the difference value between the target and the reference bands at the *nw* pixel is calculated as follows:

$$d^{nw} = A_{i-t,\,j-t} - (S_{i,\,j} + S_{i-2t,\,j-2t})/2, \tag{1}$$

where t denotes the horizontal or vertical distance between the four neighboring pixels and (i, j), and $t = 2^{x-1}$. Thus, the *nw*, *ne*, *se*, and *sw* pixels can be represented as $(i - t, j - t)$, $(i + t, j - t)$, $(i + t, j + t)$, and $(i - t, j + t)$. The difference values at the *ne*, *se*, and *sw* pixels are calculated as d^{ne}, d^{se}, and d^{sw}, respectively, in a manner similar to (1).

Then, we calculate the four directional gradients by determining the difference between the pixels relative to each direction. In Step 1, the calculation mainly use the interpolation target band and the band in which the pixel value was observed at (i, j) in the mosaicked MSI instead of the green component. The directional gradient weight at *nw* is calculated as follows:

$$w^{nw} = 1/(|A_{i-t,\,j-t} - A_{i+t,\,j+t}|| + |S_{i-2t,\,j-2t} - S_{i,\,j}| + |\tilde{S}_{i-t,\,j-t} - S_{i,\,j}| + \varepsilon), \tag{2}$$

where $\tilde{S}_{i,\,j}$ denotes the value of the band S at (i, j) calculated by bilinear interpolation, and ε is a small positive number that prevents the gradients from attaining a value of zero. Further, the gradient weights in the *ne*, *se*, and *sw* directions are calculated as w^{ne}, w^{se}, and w^{sw}, in a manner similar to (2). Then, a difference value at (i, j) is predicted by calculating the weighted average of the four difference values at the *nw*, *ne*, *se*, and *sw* pixels using these gradient weights as follows:

$$\overline{d} = \frac{w^{nw}d^{nw} + w^{ne}d^{ne} + w^{se}d^{se} + w^{sw}d^{sw}}{w^{nw} + w^{ne} + w^{se} + w^{sw}}. \tag{3}$$

Finally, the pixel value at (i, j) is obtained as follows:

$$\bar{A}_{i,\,j} = S_{i,\,j} + \overline{d}. \tag{4}$$

Next, let the position of the pixel targeted for interpolation be (p, q), as shown in Fig. 1 (*Step b*). First, the difference value between the target and the reference bands at the *n* pixel is calculated as follows:

$$d^{n} = A_{p,\,q-t} - (S_{p,\,q} + S_{p,\,q-2t})/2. \tag{5}$$

Further, the difference values at the *s*, *w*, and *e* pixels are calculated as d^{s}, d^{w}, and d^{e}, in a manner similar to (5).

Then, we calculate the four directional gradients. The directional gradient weight at n is calculated as follows:

$$w^n = 1/(|S_{p,q-2t} - S_{p,q}| + \sum_{k=1}^{t} (|M_{p,q-(t+(k-1))} - M_{p,q+(t-(k-1))}|) + \sum_{k=1}^{t} (W_k |M_{p-k,q-2t} - M_{p-k,q}|) + \sum_{k=1}^{t} (W_k |M_{p+k,q-2t} - M_{p+k,q}|) + \varepsilon) , \tag{6}$$

where $M_{p,\,q}$ denotes the pixel value at (p, q) of the mosaicked MSI, and W_k is calculated as follows:

$$W_k = \frac{\exp(-k^2/(2\sigma^2))}{2\sum_{l=1}^{t} \exp(-l^2/(2\sigma^2))}, \tag{7}$$

where σ is a parameter. Further, the gradient weights in the s, w, and e directions are calculated as w^s, w^w, and w^e, in a manner similar to (6). Then, the difference value at (p, q) is predicted by calculating the weighted average of the four difference values at the n, s, w, and e pixels using these gradient weights.

Finally, the pixel value at (p, q) is obtained in a manner similar to (4). Then, t is assigned a value of $t/2$, and the process in Step 1 is repeated while t is greater than one.

2.3 Enhancement by Using Eight Neighboring Pixels

In Step 2, each pixel is updated by using the same process as that in Step 1 and by using the eight neighboring pixels. Let the band targeted for update be A, the position of the pixel targeted for update be (u, v) as shown in Fig. 2, and the reference band be S. First, the difference value between the target and reference bands at the nw pixel is calculated as follows:

$$d^{nw} = A_{u-1,v-1} - S_{u-1,v-1}. \tag{8}$$

Further, the difference values at the other neighboring pixels are calculated as d^{ne}, d^{se}, d^{sw}, d^n, d^s, d^w, and d^e, in a manner similar to (8). In addition, the directional gradient weights are calculated using a method similar to that in Step 1 with $t = 1$, and $M_{u,\,v}$ modified to $A_{u,\,v}$. Then, a difference value at (u, v) is predicted by calculating the weighted average of the eight difference values and these gradient weights.

Finally, the pixel value at (u, v) is obtained in a manner similar to (4).

2.4 Removal of Artifacts by Applying Median Filter

In Step 3, the artifacts of the updated MSI obtained in Step 2 are removed by using a median filter. As shown in Fig. 3, first, a 1-D smoothing filter is applied to the updated MSI in the wavelength direction. The 1-D smoothing filter is defined as follows:

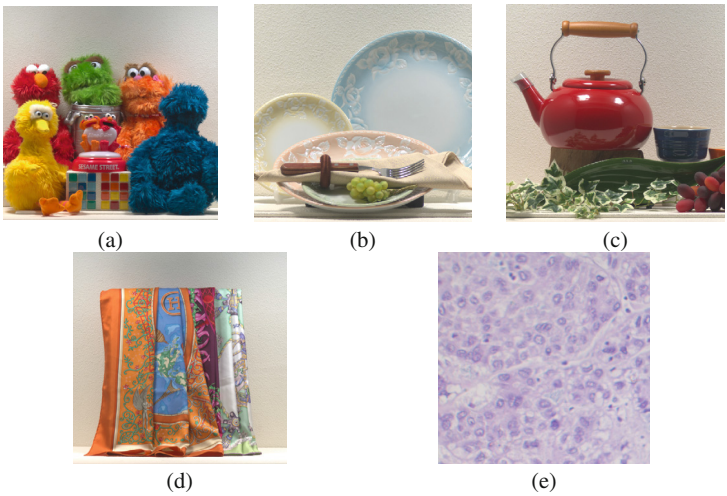

Fig. 4. Test images: (a) *Toys*, (b) *Dishes*, (c) *Kettle*, (d) *Scarf*, (e) *Pathological Image* (*PI*). The size of these images is 512 × 512 pixels.

$$f = [1/3 \quad 1/3 \quad 1/3]. \tag{9}$$

$$y = f * x, \tag{10}$$

where x is an N-band vector of the updated MSI, y is an N-band vector of the smoothed MSI, * is the convolution, and N is the number of bands of the MSI. This process is a pixel-by-pixel convolution. The one band of the smoothed MSI corresponds to the power of three bands, and we treat this value as an alternative of Y. This calculate an average among targeted band and two adjacent bands in the wavelength direction. This processing is applied to each band of MSI. For example, in the case of 16-band MSI whose bands are numbered in ascending order of their wavelengths, the average among band 1, band 2, and band 3 is calculated when the target band is band 2.

The rest of the process is almost the same as [9]. The difference between the updated MSI and smoothed MSI is calculated. Then, a 2-D median filter is applied to the calculated difference in the spatial direction. Finally, we obtain the corrected MSI by adding the smoothed MSI to the median-filtered difference.

3 Experimental Results

We show the validation of the proposed method through an experiment. In this experiment, we compare our proposed method with bilinear interpolation, the method proposed by Brauers et al. [4], the method proposed by Miao et al. [5], and Step 1 in MLDI by applying these methods to the 16-band MSIs shown in Fig. 4. In Fig. 4, the 16-band images are converted to sRGB images. The size of these images is 512 × 512 pixels. The spectral sensitivities of the camera used for capturing *Toys*, *Dishes*, *Kettle*,

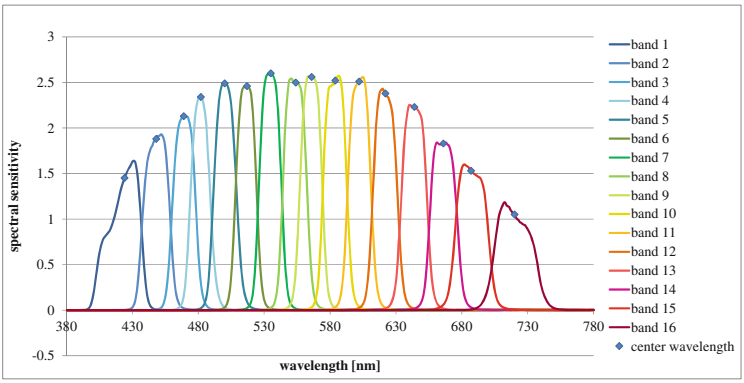

Fig. 5. The spectral sensitivities of the camera used by capturing *Toys*, *Dishes*, *Kettle*, and *Scarf*. The center wavelengths of each band are 424, 448, 469, 482, 500, 517, 535, 554, 566, 584, 602, 622, 644, 666, 687, and 720 nm. (Color figure online)

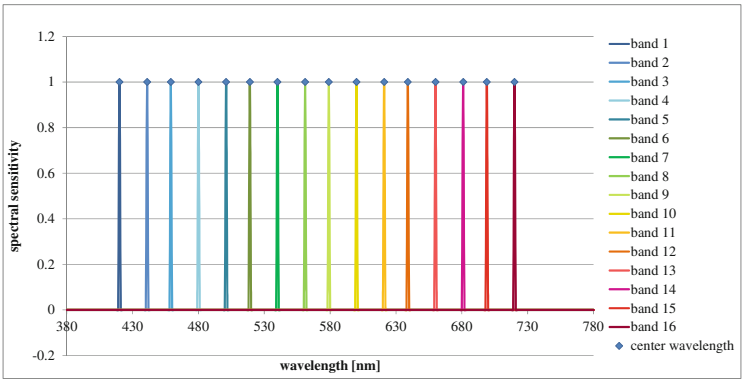

Fig. 6. The spectral sensitivities of the camera used by capturing *PI*. The center wavelengths of each band are 420, 441, 459, 480, 501, 519, 540, 561, 579, 600, 621, 639, 660, 681, 699, and 720 nm. (Color figure online)

and *Scarf* is shown in Fig. 5, and that used for capturing *PI* is shown in Fig. 6. The pathological tissue is of an H&Estained liver (US Biomax, Hepatocellular Carcinoma Tissue Array C054). The imaging system uses an optical microscope (Olympus, BX53), liquid crystal tunable filters (CRi, Varispec VIS), and monochrome CCD (Point Grey, Grasshopper 3). In MLDI, we assumed $\sigma = 0.5$ and $\varepsilon = 1$; the window size of the 2-D median filter is set to 17 × 17, and the length of the 1-D smoothing filter is set to 3. We assumed that the 16-band MSFA is arranged in 4 × 4 ($x = 2$) pixels, as shown in Fig. 7. The bands are numbered in ascending order of their wavelengths. We obtain the mosaicked MSI by sampling the pixel values from the original MSI that captured by single-camera-multi-shot systems. Next, we obtain the restored images of the mosaicked MSI by applying each demosaicking method, and then, evaluate these methods based on peak signal-to-noise ratio (PSNR) and visual observation.

Table 1 shows the PSNR of the restored images for each method. As shown in Table 1, in the case of MSFA1, our proposed method outperforms the method of Brauers et al. by 1.75 [dB], and Step 1 in MLDI by 0.96 [dB] for the average value. In addition, in the case of MSFA2, our proposed method outperforms the method of Brauers et al. by 2.93 [dB], and Step 1 in MLDI by 1.22 [dB] for the average value. However, in the case of MSFA1, the PSNR of our proposed method is lower than that of bilinear interpolation by 0.56 [dB] for PI. In the case of MSFA2, our proposed method outperforms the existing methods for all the test MSIs. This result can be related to the interpolation order of the missing pixels in Step 1. If the target band is band 1 and t is two, then, in *Step a*, the target pixel is at a position surrounded by a triangle as shown in Fig. 7. Next, in *Step b*, the target pixel is at a position surrounded by a diamond. Thus, in MSFA1, the positions of bands 11, 3, and 9 are interpolated sequentially during the interpolation of band 1. On the other hand, in MSFA2, the positions of bands 2, 3, and 4 are interpolated sequentially during the interpolation of band 1. In Step 1, when the wavelength of the target and the reference bands are close, the correlation between those bands is strong, then, interpolation accuracy is high. If the interpolation error is small when t is large, the error is also small when t is small because previously interpolated pixels are used for the interpolation. Thus, in the case

Fig. 7. MSFA pattern: (a) MSFA1, (b) MSFA2. If the interpolation target band of Step 1 is band 1 and t is two, a position surrounded by a triangle is the target pixel in *Step a*, and a position surrounded by a diamond is the target pixel in *Step b*.

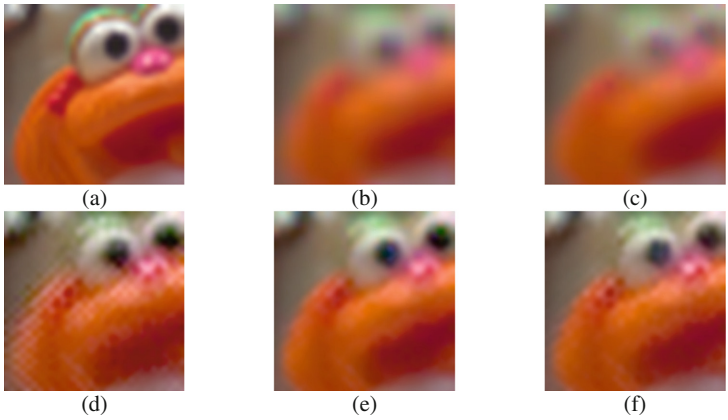

Fig. 8. Comparison between demosaicked images of *Toys* in the case of MSFA2: (a) Original, (b) Bilinear, (c) Miao et al. [5], (d) Brauers et al. [4], (e) MLDI (Step 1), (f) MLDI.

Table 1. Comparison of each method in terms of PSNR: (a) MSFA1, (b) MSFA2.

Method	*Toys*	*Dishes*	*Kettle*	*Scarf*	*PI*	*Average*
(a)						
Bilinear	29.46	27.32	26.70	28.10	**32.83**	28.88
Miao	29.61	27.64	26.87	28.20	32.60	28.98
Brauers	31.40	29.74	29.05	29.00	31.72	30.18
MLDI (Step 1)	32.19	30.88	29.81	29.78	32.26	30.99
MLDI (Step 1-3)	**33.18**	**32.35**	**31.20**	**30.63**	32.27	**31.93**
(b)						
Bilinear	29.44	27.31	26.66	28.09	32.82	28.86
Miao	29.60	27.64	26.83	28.20	32.60	28.97
Brauers	31.55	29.83	29.08	29.27	31.95	30.34
MLDI (Step 1)	32.91	31.22	30.04	31.32	34.75	32.05
MLDI (Step 1-3)	**34.43**	**32.95**	**31.74**	**32.36**	**34.86**	**33.27**

of MSFA2, interpolation accuracy is high because the wavelength of the target and the reference bands are close when t is large. Therefore, for MLDI, MSFA2 is more suitable than MSFA1.

In addition, Fig. 8 shows the sRGB versions of the images obtained by applying each method to the image *Toys* in the case of MSFA2. As shown in Fig. 8, the restored image obtained by applying MLDI has fewer artifacts than the one obtained by applying the other existing methods. This result can be attributed to the consideration of inter-channel correlation and spatial gradient in MLDI.

The proposed method has a room for improvement by changing the number of neighboring pixels of Step 2, and this is one of our future work. We use eight neighboring pixels in this step, but observed pixels are not included in the window when a pixel of band 1 is interpolated at the triangle position of Fig. 7. It has the potential to be improved by adjusting the windows size depending on the dense of the observed pixels.

In future work, it is necessary for our proposed method to consider the distance between the interpolation target and the reference bands in the wavelength direction for the interpolation in order to improve the quality of the restored image because the quality of the restored image obtained by our proposed method is poorer than the original image.

4 Conclusion

In this paper, we proposed MLDI that is based on LDI and considers inter-channel correlation and spatial gradient. We evaluated the performance of our proposed method by comparing it with existing methods. Experimental results show that our proposed method outperforms the existing methods. In future work, we plan to enhance MLDI and compare it with existing methods for various MSFAs.

Acknowledgement. This work was supported by JSPS KAKENHI Grant Number 15K20899. We would like to thank Masahiro Yamaguchi at the Tokyo Institute of Technology for providing the test images used in our experiments.

References

1. Tashiro, M., Murakami, Y., Yamaguchi, M., Obi, T., Ohyama, N., Abe, T., Yagi, Y.: Efficient implementation of dye amount adjustment in pathological images using multispectral pathological imaging. Med. Imaging Technol. **26**(4), 240–246 (2008)
2. Monno, Y., Tanaka, M., Okutomi, M.: Multispectral demosaicking using adaptive kernel upsampling. In: Proceedings of IEEE International Conference on Image Processing (ICIP), pp. 3218–3221 (2011)
3. Kopf, J., Cohen, M.F., Lischinski, D., Uyttendaele, M.: Joint bilateral upsampling. ACM Trans. Graphics **26**(3), 96-1–96-5 (2007)
4. Brauers, J., Aach, T.: A color filter array based multispectral camera. In: Proceedings of Workshop Farbbildverarbeitung (2006)
5. Miao, L., Qi, H., Ramanath, R., Snyder, W.E.: Binary tree-based generic demosaicking algorithm for multispectral filter arrays. IEEE Trans. Image Process. **15**(11), 3550–3558 (2006)
6. Monno, Y., Tanaka, M., Okutomi, M.: Multispectral demosaicking using guided filter. In: Proceedings of SPIE, vol. 8299, pp. 82990O-1–82990O-7 (2012)
7. He, K., Sun, J., Tang, X.: Guided image filtering. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part I. LNCS, vol. 6311, pp. 1–14. Springer, Heidelberg (2010)
8. Zhang, L., Wu, X., Buades, A., Li, X.: Color demosaicking by local directional interpolation and nonlocal adaptive thresholding. J. Electron. Imaging **20**(2), 023016-1–023016-16 (2011)
9. Buades, A., Coll, B., Morel, J.-M., Sbert, C.: Self-similarity driven color demosaicking. IEEE Trans. Image Process. **18**(6), 1192–1202 (2009)