

# Evaluation of a Hyperspectral Image Database for Demosaicking purposes

Mohamed-Chaker Larabi<sup>a</sup> and Sabine Süsstrunk<sup>b</sup>

<sup>a</sup> XLim Lab, Signal Image and Communication dept. (SIC)  
University of Poitiers, Poitiers, France

<sup>b</sup> School of Computer and Communication Sciences (IC)  
Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland

## ABSTRACT

We present a study on the applicability of hyperspectral images to evaluate color filter array (CFA) design and the performance of demosaicking algorithms. The aim is to simulate a typical digital still camera processing pipe-line and to compare two different scenarios: evaluate the performance of demosaicking algorithms applied to raw camera RGB values before color rendering to sRGB, and evaluate the performance of demosaicking algorithms applied on the final sRGB color rendered image. The second scenario is the most frequently used one in literature because CFA design and algorithms are usually tested on a set of existing images that are already rendered, such as the Kodak Photo CD set containing the well-known lighthouse image.

We simulate the camera processing pipe-line with measured spectral sensitivity functions of a real camera. Modeling a Bayer CFA, we select three linear demosaicking techniques in order to perform the tests. The evaluation is done using CMSE, CPSNR, s-CIELAB and MSSIM metrics to compare demosaicking results. We find that the performance, and especially the difference between demosaicking algorithms, is indeed significant depending if the demosaicking/demosaicking is applied to camera raw values as opposed to already rendered sRGB images. We argue that evaluating the former gives a better indication how a CFA/demosaicking combination will work in practice, and that it is in the interest of the community to create a hyperspectral image dataset dedicated to that effect.

**Keywords:** color filter array, CFA, digital still camera, spectral sensitivities, demosaicking, hyperspectral images

## 1. INTRODUCTION

Digital photography is a part of our daily life because many devices, such as digital still and video cameras, smartphones, webcams, etc., are available at affordable prices. The consumer electronic device industry has mostly adopted single-sensor imaging, which captures the three spectral wave-bands red, green, and blue on a single sensor. These cameras are more cost-effective and usually more compact than tri-sensor cameras, which use three sensors for full-resolution red, green, and blue scene information. Spectral selectivity on a single sensor, be it a charge-coupled device (CCD)<sup>1</sup> or a complementary metal oxide semiconductor (CMOS)<sup>2</sup> is achieved by adding a color filter array (CFA)<sup>3-5</sup> in front of the sensor, such as the most common Bayer filter.<sup>6</sup> The resulting image from CFA acquisition is a gray-scale image with one single color at each pixel. To recover tri-component or full-color at each pixel, the image needs to be processed with a color reconstruction algorithm called *demosaicking*. Figure 1 gives an example of a Bayer CFA image (a) and its demosaicked result (b).

The design of color filter arrays (CFA) and associated demosaicking algorithms is still an active research topic, as the perfect spatial arrangement of the filters and their spectral characteristics have a large influence on image quality. Image demosaicking could be regarded as an interpolation problem that creates full-resolution color images from CFA-based single-sensor images. In interpolation, the aim is to estimate the missing half amount of green pixels (quincunx interpolation) and reconstruct the missing three quarters of the red and blue

---

Further author information: (Send correspondence to M.C.L.)  
M.C.L.: E-mail: larabi@sic.univ-poitiers.fr

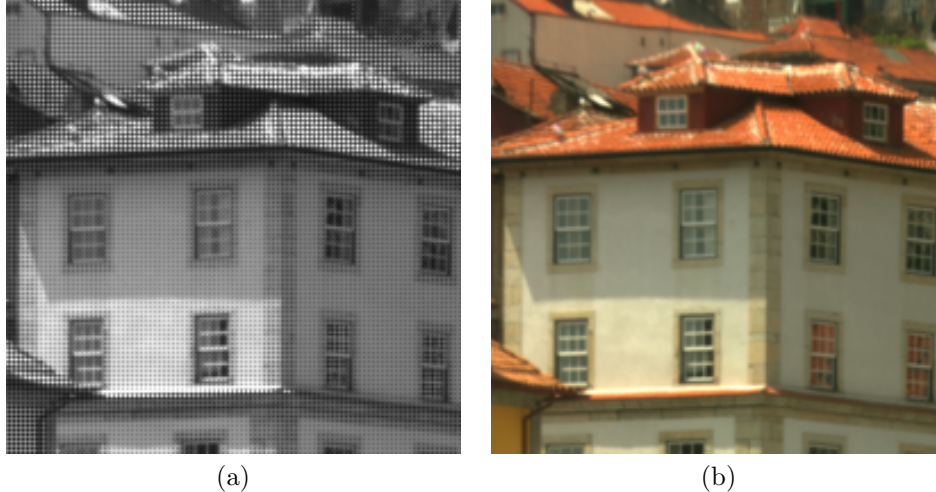


Figure 1. Bayer CFA-based single-sensor imaging: (a) grayscale CFA image and (b) full-color image.

pixels (rectangular interpolation) as performed by bilinear and bicubic interpolations. However, this approach is a simplistic view of the real demosaicking process because it does not take into account intra- and inter-channel dependencies.

Demosaicking has been studied by many researchers, we have a rich literature with various methodologies in the spatial or frequency domain, using image geometry, applying refinement and postprocessing techniques, etc. Many comprehensive comparisons and/or surveys have been published to date.<sup>7-10</sup> Although all these efforts have resulted in very efficient demosaicking algorithms, we claim that there is no appropriate image database dedicated to these types of evaluations. Many authors used and continue to use a test set composed of KODAK PhotoCD images to prove the efficiency of their demosaicking algorithm and the quality of image reconstruction. The frequency spectra of these images are very interesting, especially on the widely used lighthouse image, as they allow to easily visualize reconstruction errors. However, the original PhotoCD images are slightly compressed due to the Gaussian pyramid used in the encoding that allows extracting different resolutions from the same PCD file. Additionally, they have been rendered to a limited color gamut, similar to the current sRGB color encoding standard.<sup>11</sup> The experiments are thus conducted by recreating the CFA structure from the final rendered image without taking into account that the images have been preprocessed and that the color component information is no longer original. Li et al.<sup>10</sup> have shown that most existing demosaicking algorithms achieve good performance on the Kodak data set but their performance degrades significantly on another set they used (IMAX high-quality images with varying-hue and high-saturation edges). Their study demonstrates that for testing CFA design and demosaicking algorithms, there is a real need for new content that is adapted to the task.

In this paper, we study the applicability of the hyperspectral image database proposed by Foster et al.<sup>12</sup> for demosaicking evaluations. This database is composed of eight hyperspectral images (see Figure 4). The idea behind our experiments is to simulate as close as possible the in-camera processing steps of a digital camera. Thus, we also use real measured spectral sensitivities of a camera in order to reproduce raw sensor response values. The obtained results are compared with those obtained by mosaicking/demosaicking the rendered images, and additionally with ground truth images obtained by using the 1931 CIE color matching functions (CMF) to sample the hyperspectral data.

The remainder of this paper is organized as follows. Section 2 describes our experimental procedure in detail. Section 3 summarizes how the hyperspectral database was obtained. The selected demosaicking techniques are briefly discussed in Section 4. Metrics are an important part of evaluating demosaicking results and they are presented in Section 5. Section 6 discusses the experimental results, and Section 7 ends the article with conclusions and future work.

## 2. EXPERIMENTAL PROCEDURE

This study’s experimental procedure is summarized in the synopsis of figure 2. Our purpose is to provide an answer to whether or not evaluating mosaicking/demosaicking algorithms directly on rendered images is correct from an image quality point of view. As discussed above, most of the published research study demosaicking algorithms on already rendered images, while in most digital cameras, demosaicking occurs in raw camera RGB before color rendering. We thus simulate the relevant in-camera processing steps of a digital camera, from acquisition to rendering, and compare results obtained by applying demosaicking before and after sRGB rendering. Our simulation is similar to the one of Alleysson et al.,<sup>13</sup> who optimized camera spectral sensitivities based on one given demosaicking algorithm.<sup>8</sup> The different steps of the synopsis of Figure 2 are described below.

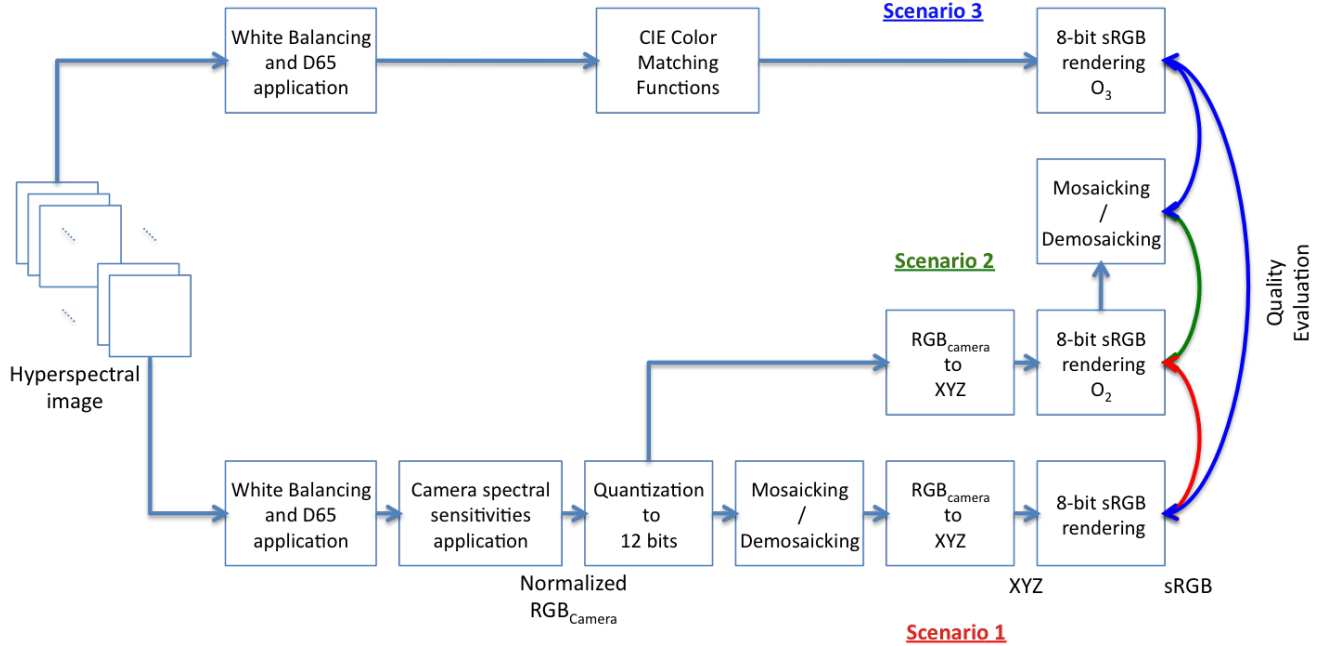


Figure 2. Synopsis of the simulations used in the paper.

Instead of acquiring real scene information from a (real) sensor, we use an already existing hyperspectral image data set that is described in Section 3. The images’ raw sensor values are obtained by applying the well known image formation model. We thus first multiply the reflectance spectra at each pixel with the spectrum of the illuminant (i.e. D65 in our case), and then multiply the resulting color signal with the spectral sensitivities. We chose D65 as the illuminant to avoid having to white-balance the image, as it is the standard illuminant used for sRGB encoding.<sup>11</sup> However, as the spectral sensitivities were derived by measuring the quantum efficiency of a real digital camera (see Figure 3-a), we do apply a gain control to each channel to compensate for the different quantum efficiencies of the red, green, and blue channels. At this stage, we have obtained an image in “camera RGB” used in the following steps.

Image formation is an analog process, and we thus calculate in floating point up to now. In order to simulate the analog to digital conversion, we perform a quantization to 12-bits per channel, which corresponds to the common coding length nowadays. The quantized image is mosaicked according to the Bayer<sup>6</sup> CFA and then demosaicked with the algorithms described in Section 4.

In order to render the mosaicked/demosaicked images to sRGB, we first need to find the linear transform that maps the pixel values from camera RGB to XYZ tristimulus values. We use a simple least squares fitting, but there are much more sophisticated methods to obtain the matrix.<sup>15</sup> Note that real camera sensitivities do not fulfill the Luther condition, in other words, they are not within a linear combination of the CIE color matching

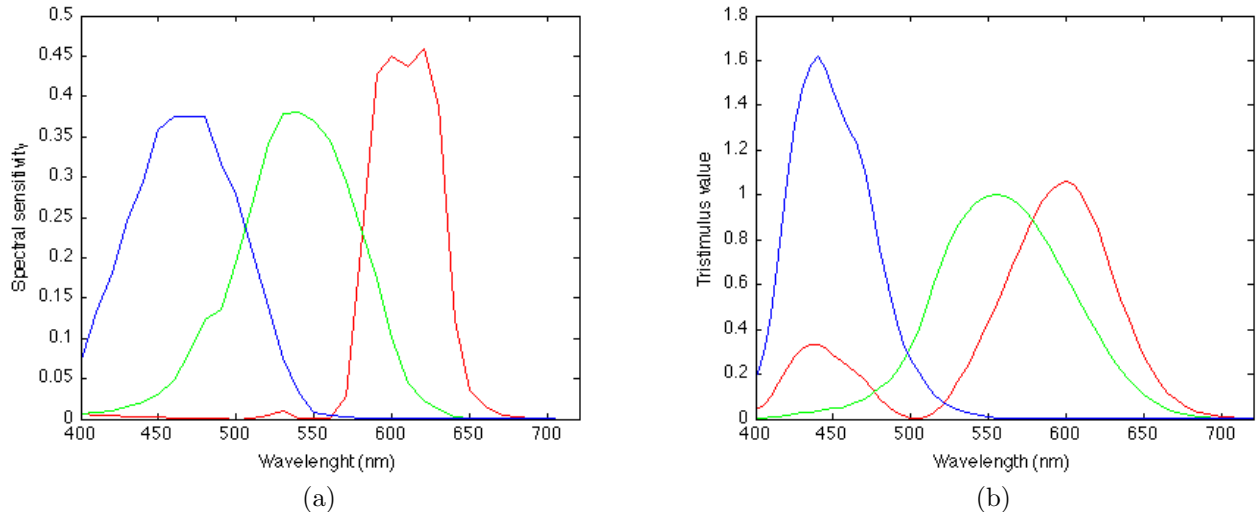


Figure 3. (a) Spectral sensitivity functions of the our digital camera and (b) the 1931 CIE Color matching functions.<sup>14</sup>

functions. As such, any linear transform will not correctly map all camera RGB values to the corresponding XYZs, and a residual error is obtained that will be reflected in the results of our simulations to the ground truth images.

After camera RGB to XYZ conversion, the images are mapped to 8-bit sRGB using the method described in the standard.<sup>11</sup> We do not consider any additional rendering operations for preferred reproduction,<sup>16</sup> as is the case in more sophisticated digital cameras that apply image specific rendering. Such additional color rendering operations will of course also influence the results, and could be considered in our simulation framework. However, as such operations are highly image and preference dependent, we omitted them in this preliminary study

The steps described above represent *scenario 1* of the synopsis given in Figure 2. Omitting the mosaicking/demosaicking step results in the "original" image, called *O2*, that we use for evaluation. For *scenario 2*, we additionally mosaic/demosaick *O2*, analogue to the procedure followed in the demosaicking literature, i.e. mosaicking/demosaicking an already rendered images (e.g., Kodak PhotoCD images). *Scenario 3* illustrates how we obtain the ground truth image, called *O3*, by simply applying the CIE color matching functions on the hyperspectral data and then rendering to sRGB. We use the image *O3* to compare the demosaicking results of scenario 1 and 2.

Note that our simulation does not include all in-camera processing, as the above mentioned white-balancing and image specific color rendering, as well as linearization, flare subtraction, noise removal and filtering, sharpening, and compression. While these are important steps that will also influence the demosaicking result, they are highly camera and/or image dependent, which makes their inclusion into a simulation very challenging.

The evaluations of the our scenarios are performed using different well-known metrics such as CMSE, CPSNR, s-CIELAB and MSSIM for structural informations. These measures are described in Section 5.

### 3. HYPERSPECTRAL DATASET

For our simulations, we use the hyperspectral image database created by Foster et al.<sup>12</sup> in 2004. For the capture of these images, the authors used a high-spatial-resolution hyperspectral imaging system to acquire data from rural and urban scenes in Portugal, namely a low-noise Peltier-cooled digital camera providing a spatial resolution of  $1344 \times 1024$  pixels (Hamamatsu, model C4742-95-12ER, Hamamatsu Photonics K.K., Japan) with a fast tunable liquid-crystal filter (VariSpec, model VS-VIS2-10-HC-35-SQ, Cambridge Research & Instrumentation, Inc., MA) mounted in front of the lens, together with an infrared blocking filter. Focal length was typically set to 75 mm and aperture to f16 or f22 to achieve a large depth of focus. The line-spread function of the system was close to

Gaussian with standard deviation approx. 1.3 pixels at 550 nm. The intensity response at each pixel, recorded with 12-bit precision, was linear over the entire dynamic range. The peak-transmission wavelength was varied in 10-nm steps over 400-720 nm. This set is composed of 8 hyperspectral images, as shown rendered to sRGB in Figure 4.



Figure 4. Rendered versions of the hyperspectral images used in our experiments.

## 4. DEMOSAICKING ALGORITHMS

As mentioned previously, there are many algorithms for single-sensor image demosaicking. Some of them are sequential, i.e. color components are interpolated separately and the others exploit inter-channel correlation. In this section, we only briefly describe the methods we use, the reader is referred to the original papers for more details. As this paper does not intend to compare demosaicking algorithms but only to evaluate the effect of using hyperspectral images with a camera simulation that includes a demosaicking step, the selected algorithms are not intended to cover the whole state-of-the-art.

### 4.1 Bilinear Model

Bilinear interpolation is one of the simplest and most used algorithm performing a high-quality linear interpolation. It interpolates a missing channel by taking the averages of the closest neighbors of the same channel. For example, the green channel at a red or blue pixel can be estimated as shown by the following equation:

$$\hat{G}(i, j) = \frac{1}{4}[G(i-1, j-1) + G(i-1, j+1) + G(i+1, j-1) + G(i+1, j+1)]. \quad (1)$$

Bilinear interpolation is perhaps the most trivial demosaicking algorithm. It completely ignores inter-channel color correlation because each channel is estimated separately. This approach offers fast demosaicking but with questionable quality, such as noticeable false colors and blur along edges.

### 4.2 Alleysson *et al.*

Alleysson *et al.*<sup>8</sup> showed that the spatial multiplexing of the red, green, and blue signal in a Bayer CFA is equivalent to multiplexing the frequency of an achromatic "luma" component and two modulated chroma components. In addition, the luminance and chrominance components are sufficiently isolated in the frequency domain to consider the construction of demosaicking algorithms based on frequency analysis. The algorithm

separately extracts estimates of luminance and modulated chrominance by filtering the Bayer CFA mosaick using two-dimensional filters with appropriate bandwidths, and then converts the estimated luma and the two demodulated chrominance values at each spatial location into RGB values.

### 4.3 Dubois *et al.*

Dubois<sup>17</sup> defined a locally-adaptive luma-chroma demultiplexing algorithm that exploits the redundancy of one chrominance in the Bayer CFA mosaick by selecting, locally, the best estimate using the more decorrelated component to the luma signal. The work in<sup>18</sup> introduced a least-squares approach for optimal filter design that replaced the window filter method used in the previous version. This new filter design method produces lower order filters that achieved virtually identical demosaicking quality as the higher order filters. We use the second method in this paper.

## 5. EVALUATION METRICS

To perform analytical assessment of the defined scenarios, we need one or several quality measure to assess different types of artifacts. We selected the commonly used CMSE and CPSNR, s-CIELAB, which characterizes the regions in the test image that are visually different from the original image, and MSSIM that measures differences in structural content. Recall that  $O_2$  and  $O_3$  are the "original" images without demosaicking, obtained by either applying camera sensitivities or color matching functions, respectively. Thus, the metrics are applied to evaluate the demosaicking results of: 1) Scenario 1 in comparison to  $O_2$ ; 2) Scenario 2 in comparison to  $O_2$ ; 3) Scenario 1 in comparison to  $O_3$ ; ad 4) Scenario 2 in comparison to  $O_3$ . These metrics are briefly described below.

### 5.1 CMSE & CPSNR

In the demosaicking literature, it is very common to use the composite peak-signal-to-noise ratio (CPSNR) to compare the reconstructed images to full color RGB images. We use equation 2 and 3 to calculate the CPSNR of a demosaicked image compared to the original one. Here,  $I(i, j, k)$  is the pixel intensity at location  $(i, j)$  of the  $k$ -th color component of the original image and  $I'(i, j, k)$  of the reconstructed image.  $M$  and  $N$  are the height and the width of the frame.

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

where

$$CMSE = \frac{1}{3 \times MN} \sum_{k=1}^3 \sum_{i=1}^M \sum_{j=1}^N [I(i, j, k) - I'(i, j, k)]^2. \quad (3)$$

### 5.2 s-CIELAB

s-CIELAB was proposed by Zhang and Wandell<sup>19</sup> as a spatial extension to CIELAB to account for how spatial pattern influences color appearance and color discrimination. The spatial extension is accomplished by performing a pre-processing on the CIELAB channels before applying the color difference formula. In our application, the input image is first converted to an opponent encoding (one luminance and two chrominance color components). Each component is spatially filtered to mimic the spatial sensitivity of the human eye. The final filtered images are then transformed into XYZ so that the standard CIELAB color difference formula can be applied.

### 5.3 MSSIM

The Multiscale Structural Similarity Index (MS-SSIM)<sup>20</sup> attempts to model the physical properties of the HVS. The MS-SSIM follows a top-down paradigm that first decomposes images into several scales and then measures contrast and structure in each scale. In addition, the luminance of the lowest scale is also measured. Finally, all the data is pooled into a single score. MS-SSIM has the advantage that it is computationally tractable while still providing reasonable correlations to subjective measurements.

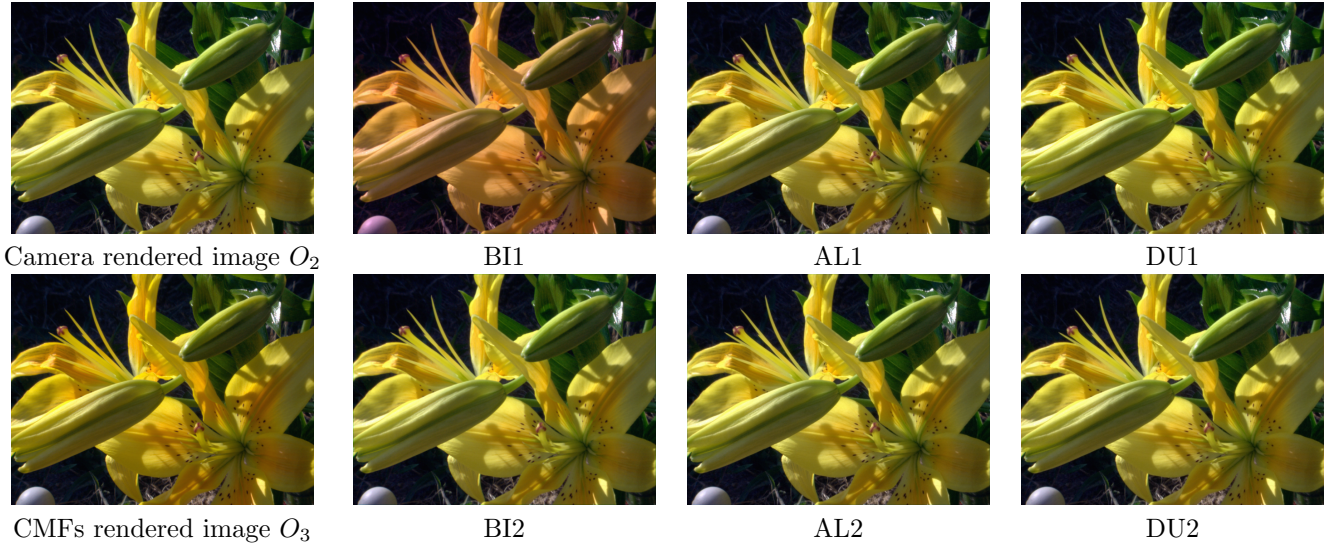


Figure 5. Results for the scene S4 from the hyperspectral database.  $O_2$  and  $O_3$  are the "original" images rendered in scenario 2 and scenario 3, respectively. BI1, AL1, and DU1 are Bilinear, Alleysson, and Dubois results for scenario 1, respectively. BI2, AL2, and DU2 are Bilinear, Alleysson, and Dubois results for scenario 2, respectively.

## 6. RESULTS AND DISCUSSION

Here, we evaluate the results of the scenarios described in Section 2 and illustrated in Figure 2. Recall that scenario 1 is the simulation of a camera pipe-line (demosaicking before sRGB rendering). In scenario 2, we perform demosaicking after color rendering (as done in the literature). The mosaicking/demosaicking step corresponds to creating mosaicks according to the Bayer CFA<sup>6</sup> before applying the three different demosaicking algorithms. For scenario 3, we generate a ground truth image obtained by applying the CIE color matching functions to the hyperspectral data and rendering directly to sRGB.

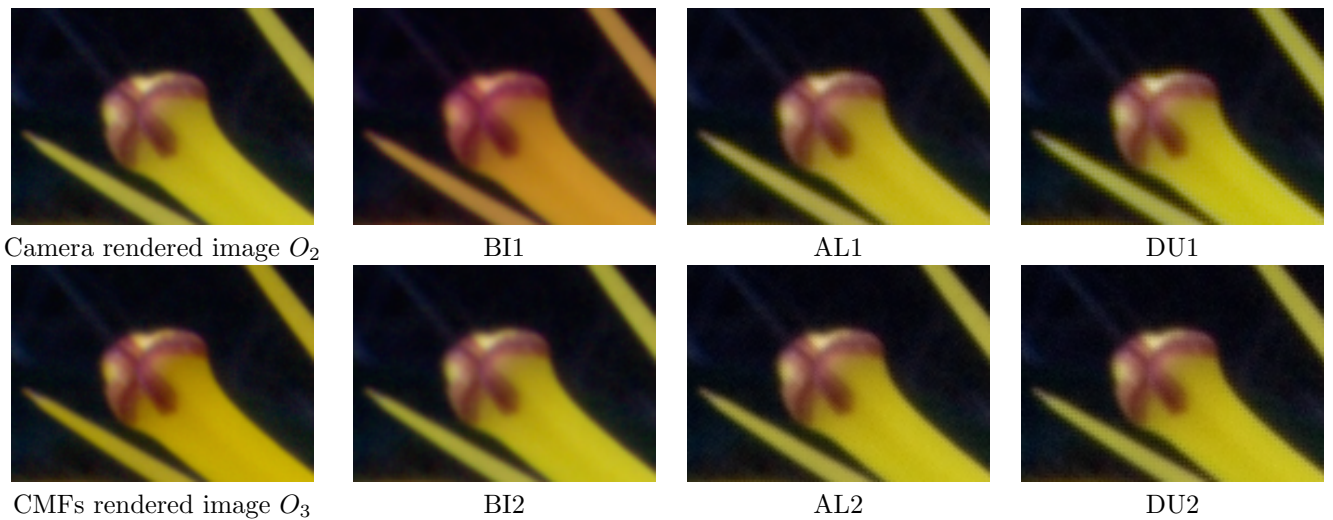


Figure 6. A zoom on part of scene S4.  $O_2$  and  $O_3$  are the "original" images rendered in scenario 2 and scenario 3, respectively. BI1, AL1, and DU1 are Bilinear, Alleysson, and Dubois results for scenario 1, respectively. BI2, AL2, and DU2 are Bilinear, Alleysson, and Dubois results for scenario 2, respectively.

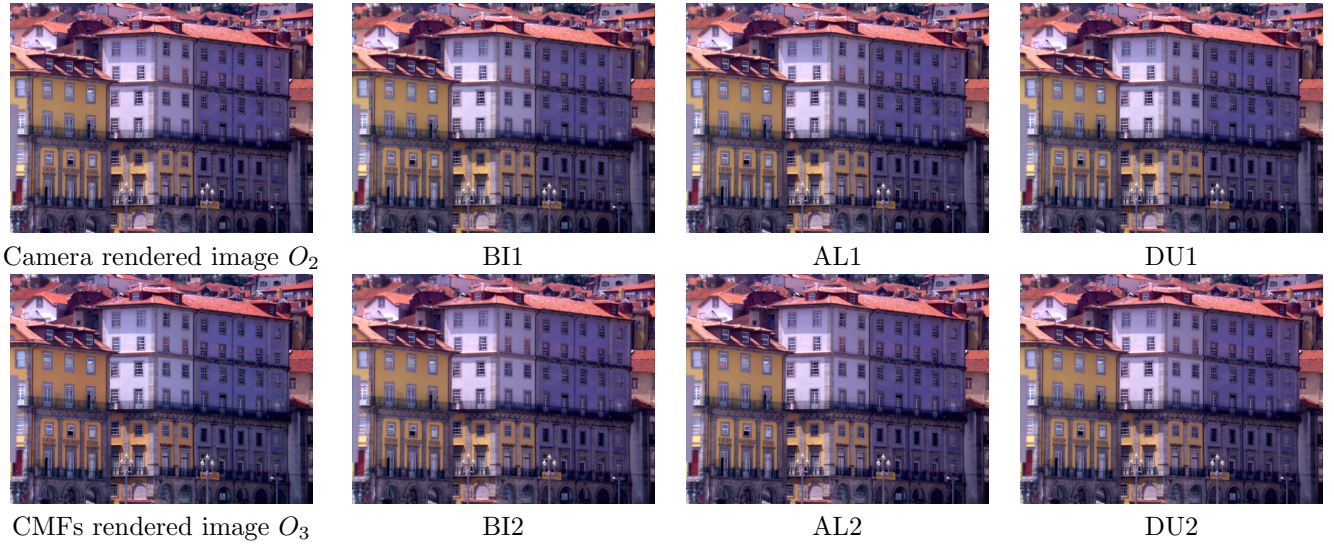


Figure 7. Results for the scene S7 from the hyperspectral database.  $O_2$  and  $O_3$  are the “original” images rendered in scenario 2 and scenario 3, respectively. BI1, AL1, and DU1 are Bilinear, Alleysson, and Dubois results for scenario 1, respectively. BI2, AL2, and DU2 are Bilinear, Alleysson, and Dubois results for scenario 2, respectively.

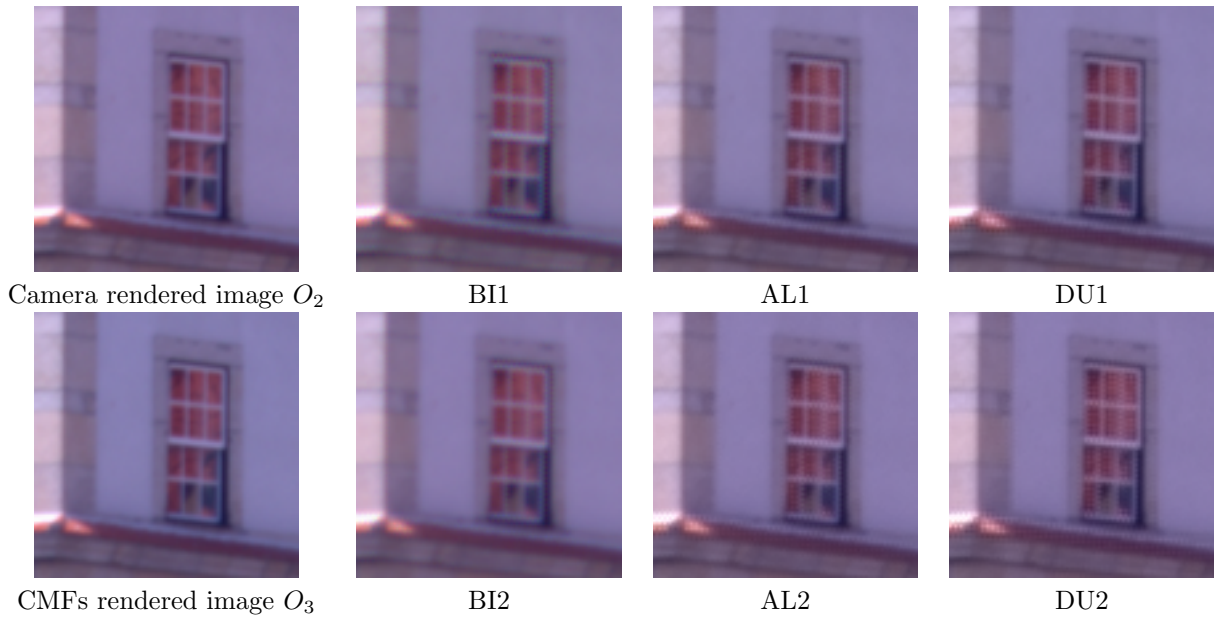


Figure 8. A zoom on part of scene S7.  $O_2$  and  $O_3$  are the “original” images rendered in scenario 2 and scenario 3, respectively. BI1, AL1, and DU1 are Bilinear, Alleysson, and Dubois results for scenario 1, respectively. BI2, AL2, and DU2 are Bilinear, Alleysson, and Dubois results for scenario 2, respectively.



For each scene  $S^*$ , except  $S5$ , we executed the three scenarios, thus obtaining the originals of scenario 2 and 3 ( $O_2^{S^*}$  and  $O_3^{S^*}$ ) and the results of the selected demosaicking algorithms for scenario 1 and 2 ( $BI1^{S^*}$ ,  $AL1^{S^*}$ ,  $DU1^{S^*}$ ,  $BI2^{S^*}$ ,  $AL2^{S^*}$  and  $DU2^{S^*}$ ). Scene  $S5$  was rejected from the experiments because it doesn't contain the same number of spectral bands than the 7 others. Figures 5 and 7 show the results obtained with all scenarios for scene  $S4$  and  $S7$ . We can notice that the rendered images are close to those given by Foster et al. (see Figure 4) except for a color difference because they manually edited the pictures. It is difficult to visually evaluate the difference between the results except for bilinear interpolation. The latter gives very different results when applied in scenario 1 or 2.

In order to better perceive the artifacts generated, we zoom into parts of scenes  $S4$  (Fig. 6) and  $S7$  (Fig. 8). We note some demosaicking artifacts around the pistil of the flower in  $S4$  and around the window in  $S7$ . Also, there seems to be more artifacts in the images from scenario 2 than in those from scenario 1, especially around the windows. However, that result is not corroborated by the objective metric s-CIELAB in Table 1 and 2, which is supposed to predict visual differences.

Table 1. Evaluation of scenario 1 (BI1, AL1, DU1) and scenario 2 (BI2, AL2, DU2) demosaicking results in comparison to the original image obtained in scenario 2 ( $O_2$ ) using CMSE, CPSNR, s-CIELAB and MSSIM. BI1, AL1, and DU1 are Bilinear, Alleysson, and Dubois results for scenario 1, respectively. BI2, AL2, and DU2 are Bilinear, Alleysson, and Dubois results for scenario 2, respectively.

Scene	Measure	BI1	AL1	DU1	BI2	AL2	DU2
S1	CMSE	121,22	31,07	37,47	21,68	23,79	27,03
	CPSNR	27,30	33,21	32,39	34,77	34,37	33,81
	s-CIELAB	6,18	2,84	2,69	0,83	1,45	1,43
	MSSIM	0,9827	0,9880	0,9853	0,9934	0,9890	0,9862
S2	CMSE	233,98	54,06	73,76	28,17	46,76	56,61
	CPSNR	24,44	30,80	29,45	33,63	31,43	30,60
	s-CIELAB	13,74	4,36	5,20	1,47	2,42	2,48
	MSSIM	0,9787	0,9851	0,9799	0,9935	0,9871	0,9837
S3	CMSE	63,45	47,28	53,54	13,68	20,76	25,13
	CPSNR	30,11	31,38	30,84	36,77	34,96	34,13
	s-CIELAB	4,15	2,28	2,77	0,84	1,33	1,36
	MSSIM	0,9864	0,9700	0,9686	0,9930	0,9877	0,9853
S4	CMSE	265,60	19,68	26,38	8,34	11,06	12,54
	CPSNR	23,89	35,19	33,92	38,92	37,69	37,15
	s-CIELAB	18,97	1,83	2,01	0,52	0,70	0,77
	MSSIM	0,9787	0,9809	0,9741	0,9921	0,9872	0,9846
S6	CMSE	187,91	17,31	17,70	13,15	8,12	10,19
	CPSNR	25,39	35,75	35,65	36,94	39,04	38,05
	s-CIELAB	12,25	3,55	3,70	0,77	0,79	0,91
	MSSIM	0,9904	0,9960	0,9951	0,9961	0,9962	0,9947
S7	CMSE	28,53	12,31	13,64	15,05	9,39	10,95
	CPSNR	33,58	37,23	36,78	36,36	38,40	37,74
	s-CIELAB	3,31	1,91	1,76	0,78	0,82	0,93
	MSSIM	0,9927	0,9957	0,9940	0,9953	0,9961	0,9949
S8	CMSE	16,09	3,56	3,94	10,68	4,09	3,70
	CPSNR	36,07	42,62	42,18	37,84	42,01	42,45
	s-CIELAB	1,64	0,53	0,63	0,53	0,44	0,45
	MSSIM	0,9940	0,9974	0,9969	0,9962	0,9975	0,9969
Average	CMSE	130,97	26,47	32,35	15,82	17,71	20,88
	CPSNR	28,68	35,17	34,46	36,46	36,84	36,28
	s-CIELAB	8,61	2,47	2,68	0,82	1,14	1,19
	MSSIM	0,9862	0,9876	0,9848	0,9942	0,9915	0,9895

For the quantitative evaluation, we used the metrics described in section 5, i.e. CMSE, CPSNR, s-CIELAB and MSSIM. These metrics have been calculated between demosaicked images of scenario 1 and 2 ( $BI1^{S^*}$ ,  $AL1^{S^*}$ ,  $DU1^{S^*}$ ,  $BI2^{S^*}$ ,  $AL2^{S^*}$  and  $DU2^{S^*}$ ) and the camera rendered image  $O_2^{S^*}$  (see Table 1) and the CMF rendered image  $O_3^{S^*}$  (see Table 2), respectively. By applying all these different metrics, we evaluate different types of artifacts. The CMSE and CPSNR focus on color signal differences, s-CIELAB aims at detecting perceived errors, and MSSIM evaluates the structural content of the image.

The first remark that concerns both tables is that the results are highly dependent on image content, arguing that a large image data set should be available as an analysis based on average performance might not be meaningful. Additionally, these results also argue for a common image data set to evaluate different algorithms,

such as is available with the Kodak images. However, both tables clearly show that there is a difference between scenario 1 and scenario 2 with regards to performance. In general, the more realistic camera processing, as simulated with scenario 1, results in worse performance than the usually applied scenario 2. This is similar to what was found by Li et al.<sup>10</sup> when applying mosaicking/demosaicking to IMAX images. This argues for a more realistic simulation to evaluate such algorithms.

Among the selected techniques, bilinear interpolation is the worst for scenario 1 and with the highest difference (up to a CPSNR of 14 dB for S4, for instance). Alleysson et al.'s technique is performing better than Dubois for all the images, but with a higher difference in scenario 1. Thus, the difference in performance of algorithms is better evaluated with a simulation that is closer to a real camera pipe-line as opposed to what is currently done (i.e. scenario 2).

As expected, all errors are much higher when comparing the performance to the CMF rendered image  $O_3^{S*}$  (see Table 2). When only evaluating the influence of mosaicking/demosaicking, thus assuming the other processing parameters remain the same, it is thus probably more appropriate to use  $O_2^{S*}$  as the ground truth to compare with. Table 3 evaluates the difference between  $O_2^{S*}$  and  $O_3^{S*}$ , using the same metrics, for all 7 scenes. Note that the difference can be very high as for the case of S2 and S4. Additionally, it is more difficult to discriminate the demosaicking techniques between scenario 1 and scenario 2, the difference is smaller. Thus, comparing with the ground truth images tends to compress the difference between the demosaicking algorithms, which in the case of this study is coherent with the visual results of figure 5 and 7. It may be suitable to compare the output of a simulated camera pipe-line like scenario 1 with the original image of scenario 3 to get a better visual judgment. This can be confirmed with a psychophysical experiment.

Table 2. Evaluation of scenario1 (BI1, AL1, DU1) and scenario2 (BI2, AL2, DU2) demosaicking results in comparison to the original image obtained in scenario 3 ( $O_3$ ) using CMSE, CPSNR, s-CIELAB and MSSIM. BI1, AL1, and DU1 are Bilinear, Alleysson, and Dubois results for scenario 1, respectively. BI2, AL2, and DU2 are Bilinear, Alleysson, and Dubois results for scenario 2, respectively.

Scene	Measure	BI1	AL1	DU1	BI2	AL2	DU2
S1	CMSE	95,66	54,57	65,82	50,37	50,74	55,50
	CPSNR	28,32	30,76	29,95	31,11	31,08	30,69
	s-CIELAB	4,49	4,54	5,10	4,96	5,25	5,19
	MSSIM	0,9809	0,9838	0,9795	0,9873	0,9837	0,9795
S2	CMSE	192,27	147,69	234,82	170,85	187,84	199,40
	CPSNR	25,29	26,44	24,42	25,80	25,39	25,13
	s-CIELAB	14,64	8,00	10,04	10,18	10,43	10,50
	MSSIM	0,9711	0,9738	0,9658	0,9820	0,9756	0,9701
S3	CMSE	44,27	70,95	66,52	45,04	49,25	55,45
	CPSNR	31,67	29,62	29,90	31,59	31,21	30,69
	s-CIELAB	4,76	3,51	3,83	2,48	2,74	2,78
	MSSIM	0,9830	0,9708	0,9680	0,9872	0,9848	0,9815
S4	CMSE	231,89	152,93	174,89	132,15	136,93	137,82
	CPSNR	24,48	26,29	25,70	26,92	26,77	26,74
	s-CIELAB	13,98	5,24	7,71	6,10	6,08	6,13
	MSSIM	0,9407	0,9341	0,9294	0,9508	0,9443	0,9420
S6	CMSE	183,47	25,15	30,62	25,80	21,56	24,16
	CPSNR	25,50	34,13	33,27	34,01	34,79	34,30
	s-CIELAB	10,84	3,23	5,26	3,25	3,13	3,17
	MSSIM	0,9891	0,9934	0,9918	0,9929	0,9927	0,9907
S7	CMSE	56,23	40,46	39,63	37,27	32,23	33,38
	CPSNR	30,63	32,06	32,15	32,42	33,05	32,90
	s-CIELAB	4,76	3,64	3,38	2,91	2,91	2,94
	MSSIM	0,9874	0,9902	0,9884	0,9901	0,9906	0,9890
S8	CMSE	19,04	7,78	9,92	16,13	9,17	9,12
	CPSNR	35,33	39,22	38,17	36,05	38,51	38,53
	s-CIELAB	2,70	1,78	2,17	1,87	1,88	1,90
	MSSIM	0,9923	0,9960	0,9954	0,9943	0,9960	0,9952
Average	CMSE	117,55	71,36	88,89	68,23	69,67	73,55
	CPSNR	28,75	31,22	30,51	31,13	31,54	31,28
	s-CIELAB	8,03	4,28	5,36	4,54	4,63	4,66
	MSSIM	0,9778	0,9774	0,9740	0,9835	0,9811	0,9783

Table 4 shows the average correlation between the four metrics used in our evaluation, for the seven scenes and the measures listed in Table 1 and Table 2. We can thus evaluate whether the metrics give consistent

Table 3. Evaluation of the difference between  $O_2$  (original image obtained from scenario 2) and  $O_3$  (original image obtained from scenario 3).

Metrics	S1	S2	S3	S4	S6	S7	S8	Average
CMSE	31,34	143,44	30,93	126,10	13,30	22,66	5,14	53,27
CPSNR	33,17	26,56	33,23	27,12	36,89	34,58	41,02	33,23
s-CIELAB	4,89	10,41	2,26	6,01	3,00	2,77	1,85	4,46
MSSIM	0,9931	0,9885	0,9945	0,9557	0,9964	0,9948	0,9983	0,9888

results. From the correlation of Table 1, the high values between CMSE and CPSNR are not surprising because the computation of the second depends on the first. However, the high correlation between s-CIELAB and CPSNR (0.96) and thus CMSE (0.98) was not expected, especially because these measures do not focus on the same artifacts as stated before. Finally, the results of MSSIM are not highly correlated with the others because it focuses mainly on the structure of the image. That being said, the correlation ratio is high enough to say that all the metrics indicate similar performance. These observations do not hold for the results in Table 2. The correlation between s-CIELAB and CPSNR (CMSE) decreases drastically with losses around 30%. It is also lower for MSSIM but the decrease is less, around 7%. This again argues against using scenario 3 for comparison.

## 7. CONCLUSION

We studied the use of hyperspectral images for the purpose of single-sensor image demosaicking evaluation. We designed an in-camera processing pipe-line to render the hyperspectral data to sRGB images. We could thus evaluate the performance of demosaicking algorithms applied to raw camera RGB values (scenario 1), which is closer to real camera design, and to compare with current evaluation practices that evaluate demosaicking on already rendered images (scenario 2).

We demonstrated the usefulness of using scenario 1 by comparing three different demosaicking algorithms and evaluating the reconstruction results with four different metrics (CMSE, CPSNR, s-CIELAB, and MSSIM). We found that in general, the demosaicking algorithms perform worse in scenario 1 than scenario 2. Additionally, the differences between the algorithms are more evident in scenario 1. We thus conclude that scenario 1, which is closer to real in-camera processing, provides a more accurate evaluation of demosaicking than current practices, which is to evaluate on already color rendered images. However, to implement scenario 1, we need a hyperspectral image data set.

Even though Foster et al. did our community a service by creating the hyperspectral image database and making it freely available, these images are not adapted for demosaicking evaluation purposes. This is partly due to the low-pass behavior of the optics of the real camera used to capture the images, which filters the high frequencies that often create problems for demosaicking algorithms, but that facilitates visual judgement. Additionally, there is not enough variation in scene content and chromaticity to be a representative sample of the world.

In this contribution, we focused only on the evaluation of demosaicking algorithms. The same framework can of course also be applied to study joint color filter array design/demosaicking. Thus, it is of benefit to the community to build a new hyperspectral database specific to this purpose by selecting scenes like the famous "lighthouse" from Kodak PhotoCD, which has image characteristics that facilitate the visual interpretation of the algorithms' performance.

Table 4. Correlation between evaluation metrics.

Metrics	Correlation of Table 1				Correlation of Table 2			
	CMSE	CPSNR	s-CIELAB	MSSIM	CMSE	CPSNR	s-CIELAB	MSSIM
CMSE	1	-0,983	0,961	-0,714	1	-0,989	0,645	-0,679
PSNR	x	1	-0,980	0,807	x	1	-0,673	0,739
s-CIELAB	x	x	1	-0,781	x	x	1	-0,603
MSSIM	x	x	x	1	x	x	x	1

## ACKNOWLEDGMENTS

A special thanks to Foster *et al.* for making their hyperspectral image database<sup>12</sup> freely available.

## REFERENCES

- [1] Dillon, P. L. P., Lewis, D. M., and Kaspar, F. G., “Color imaging system using a single CCD area array,” *IEEE Journal of Solid-State Circuits* **13**(1), 28–33 (1978).
- [2] Lule, T., Benthien, S., Keller, H., Mutze, F., Rieve, P., Seibel, K., Sommer, M., and Bohm, M., “Sensitivity of CMOS based imagers and scaling perspectives,” *IEEE Transactions on Electron Devices* **47**(11), 2110–2122 (2000).
- [3] Lukac, R. and Plataniotis, K. N., “Color filter arrays: Design and performance analysis,” *IEEE Transactions on Consumer Electronics* **51**(4), 1260–1267 (2005).
- [4] Li, Y., Hao, P., Lin, Z., Li, Y., Hao, P., and Lin, Z., “Color filter arrays: representation and analysis,” tech. rep. (2008).
- [5] Lu, Y. M., Fredembach, C., Vetterli, M., and Susstrunk, S., “Designing color filter arrays for the joint capture of visible and near-infrared images,” in [*IEEE ICIP 2009*], 3797–3800 (2009).
- [6] Bayer, B., “Color imaging array,” (1976). US Patent 3.971.065, Eastman Kodak Company, Patent and Trademark Office, Washington, D.C.
- [7] Ramanath, R., Snyder, W. E., Bilbro, G. L., and Sander, W. A., “Demosaicking methods for bayer color arrays,” *Journal of Electronic Imaging* **11**(3), 306315 (2002).
- [8] Alleysson, D., Susstrunk, S., and Herault, J., “Linear demosaicing inspired by the human visual system,” *IEEE Transactions on Image Processing* **14**(4), 439–449 (2005).
- [9] Gunturk, B. K., Glotzbach, J., Altunbasak, Y., Schafer, R. W., and Mersereau, R. M., “Demosaicking: color filter array interpolation,” *IEEE Signal Processing Magazine* **22**(1), 4454 (2005).
- [10] Li, X., Gunturk, B. K., and Zhang, L., “Image demosaicing: a systematic survey,” in [*Proc. IS&T/SPIE Conf. on Visual Communication and Image Processing*], **6822** (2008).
- [11] IEC 61966 2-1:1999, “Multimedia systems and equipment - colour measurement and management - Part 2-1: colour management-default RGB colour space - sRGB,” (1999).
- [12] Foster, D. H., Nascimento, S. M. C., and Amano, K., “Information limits on neural identification of coloured surfaces in natural scenes,” *Visual Neuroscience* **21**, 331–336 (2004).
- [13] Alleysson, D., Susstrunk, S., and Marguier, J., “Influence of Spectral Sensitivity Functions on color demosaicing,” in [*Proceedings IS&T/SID 11th Color Imaging Conference*], **11**, 351–357 (2003).
- [14] CVRL, “Cie color matching functions data sets,” (1988).
- [15] Finlayson, G. and Drew, M., “White-point preserving color correction,” in [*IS&T/SID Color Imaging Conference*], 258–261 (1997).
- [16] ISO 22028-1:2004, “Photography and graphic technology extended colour encodings for digital image storage, manipulation and interchange part 1: Architecture and requirements,” (2004).
- [17] Dubois, E., “Frequency-domain methods for demosaicking of bayer-sampled color image,” *IEEE Signal Processing Letters* **12**, 847850 (2005).
- [18] Dubois, E., “Filter design for adaptive frequency-domain bayer demosaicking,” in [*IEEE International Conference on Image Processing*], 27052708 (2006).
- [19] Zhang, X. M. and Wandell, B. A., “A spatial extension of cielab for digital color image reproduction,” *SID Journal* **5**(1), 61–63 (1997).
- [20] Z. Wang, E.P. Simoncelli and A.C. Bovik, “Multi-scale structural similarity for image quality assessment,” *IEEE Asilomar Conf. on Signals, Systems and Computers* (2003).