HAL
archives-ouvertes.fr

Practical Implementation of LMMSE Demosaicing Using Luminance and Chrominance Spaces

Brice Chaix de Lavarène, David Alleysson, Jeanny Hérault

▶ To cite this version:

HAL Id: hal-00157708
https://hal.archives-ouvertes.fr/hal-00157708

Submitted on 26 Jun 2007

Practical Implementation of LMMSE Demosaicing Using Luminance and Chrominance Spaces.

Brice Chaix de Lavarène [*,1], David Alleysson [2], Jeanny Hérault [1]

Abstract

Most digital color cameras sample only one color at each spatial location, using a single sensor coupled with a color filter array (CFA). An interpolation step called demosaicing (or demosaicking) is required for rendering a color image from the acquired CFA image. Already proposed Linear Minimum Mean Square Error (LMMSE) demosaicing provides a good tradeoff between quality and computational cost for embedded systems. In this paper we propose a modification of the stacked notation of superpixels, which allows an effective computing of the LMMSE solution from an image database. Moreover, this formalism is used to decompose the CFA sampling into a sum of a luminance estimator and a chrominance projector. This decomposition allows interpreting estimated filters in term of their spatial and chromatic properties and results in a solution with lower computational complexity than other LMMSE approaches for the same quality.

Key words: demosaicing, demosaicking, interpolation, color, luminance, chrominance, Wiener filtering

1 Introduction

A color image can be represented as a vector of three components per pixel, measuring the light intensity in three wavelength bands of the visible spectrum (Red, Green and Blue), recalling the trichromacy of human color vision (see [1] for a review on vector image processing). In order to reduce cost and size, most

* brice.chaix@lis.inpg.fr
[1] B. Chaix de Lavarène and J. Hérault are with the Images and Signals Laboratory (LIS), Joseph Fourier University, Grenoble, France.
[2] D. Alleysson is with the Psychology and Neurocognition Laboratory (LPNC), Pierre Mendès France University, Grenoble, France.
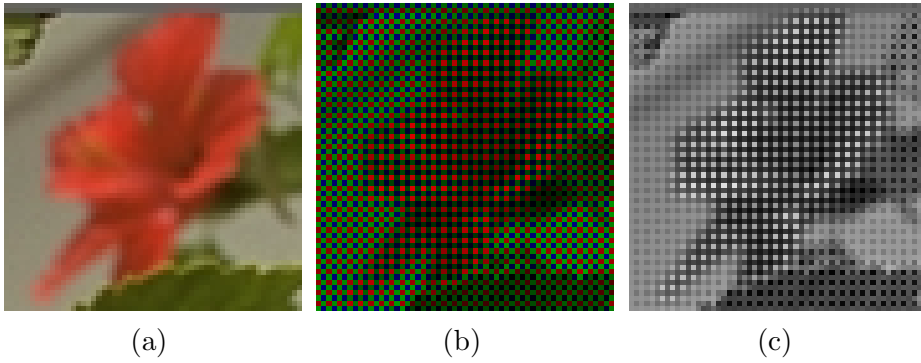
Fig. 1. (a) a color image with R, G and B values at each pixel, (b) a color CFA image with a single chromatic value per pixel according to the Bayer CFA pattern [6], (c) the corresponding grayscale CFA image as captured by the device.

digital cameras use a single sensor. The sensor is then coupled to a color filter array (CFA) and samples only one chromatic value at each spatial location as shown in Figure 1. The digital values captured by the sensor are shown in Figure 1(c). These digital values correspond to intensity levels of different colors as shown in Figure 1(b), which are the subsampled RGB values of the color image in Figure 1(a). The captured image - called a CFA image - needs to be interpolated in order to retrieve a color image with three components per pixel [2]. This operation is called demosaicing (or demosaicking [3]) and has monopolized many researchers in the field of image and signal processing from the early 80's.

Demosaicing can be seen as an inverse problem. However, considering a CFA image as a spatial multiplexing of subsampled color planes theoretically prevents of finding a solution as the inverse of the acquisition process. The subsampling operation is equivalent to a projection of the color image into a set of images having zeros according to the subsampling pattern (Figure 1(b) is actually an RGB image where the missing chromatic values are filled with zeros). In a general case, a projection is not reversible, due to the infinity of solutions for a given projected image. The recovery of the original image is only possible if we *constrain* the set of solutions. As an example, the bilinear interpolation [4,5] of each R, G and B channel isolated from the mosaic can be seen as inverting the projection onto a constrained set, where the Fourier spectrum of the elements has a shape in $(1 + cos(2\pi f))/2$ function. This cost effective solution generally generates demosaicing artifacts (blurring, false colors) because the image model is too restrictive. All demosaicing algorithms can be seen as trying to find the optimal constrained set (see Section 4.1).

In the present paper, the constraint is given through Wiener filters, which are optimal in Linear Minimum Mean Square Error (LMMSE) reconstruction. In the pioneer work of Taubman [7] and Trussell [3], the bases and theoritical framework of Wiener demosaicing is given from an image model. We propose

a simplified formulation of the Wiener demosaicing without considering all the optical parameters given in the original papers [3,7] (Section 2). This allows to design a luminance/chrominance decomposition through the demosaicing process, which results in a less complex method (Section 3). To illustrate the method, we use an image database on which we perform the LMMSE estimate and measure the performance (Section 4). The presented method is competitive among others in term of quality of reconstruction and speed. It is not the best one, nevertheless it is efficient and it helps understanding the estimated filters in term of their spatial and chromatic properties.

2 Linear Wiener demosaicing

In [3,7], the use of stacked notation unfolds the hyperstectral image of size $H \times W \times P$ into a column vector of size $HWP \times 1$, where H,W and P are respectively the height, the width and the number of spectral bands of the image. This allows expressing the model of image formation as a matrix multiplication between the hyperspectral original image, a blurring matrix (optical behavior), a spectral sensitivities matrix and the sampling matrix. In the present article we will only consider the process where $\mathbf{Y}$ is a color image with three components per pixel ($P = 3$) and $\mathbf{X}$ is a CFA image:

$$\mathbf{X} = \mathbf{PrY} \tag{1}$$

with $\mathbf{Pr}$ being a projection operator that represents the sampling process converting the image with three colors per pixels to a CFA image. Both $\mathbf{X}$ and $\mathbf{Y}$ are random variables. Taubman [7] introduced in his paper the concept of "superpixel." A superpixel is a group of pixels that matches the basic pattern of the CFA. In the Bayer CFA the basic pattern is composed of four pixels arranged on a 2×2 square: one red, two green and one blue (Figure 2). At the scale of the superpixel, the mosaic is regular, a tiling of superpixels. With the assumption that the acquisition process is invariant over the image, which is widely used, it allows the design of space-invariant filters at this scale, or in other words of block shift-invariant filters [8]. In Portilla *et al.* [9], the change over the basic pattern of the CFA is done by estimating four filters.

But while Taubman and Trussell stacked the color image $\mathbf{Y}$ into a $12HW \times 1$ column vector that imposed a large $3HW \times 12HW$ sampling matrix, we have chosen to stack the superpixels row-wise into a $12 \times \frac{HW}{4}$ matrix, as shown in Figure 2. This allows the use of a 4×12 sampling matrix $\mathbf{Pr}$, which is much more convenient to handle.

The goal of linear demosaicing is to find a matrix $\mathbf{D}$ that will recover the color
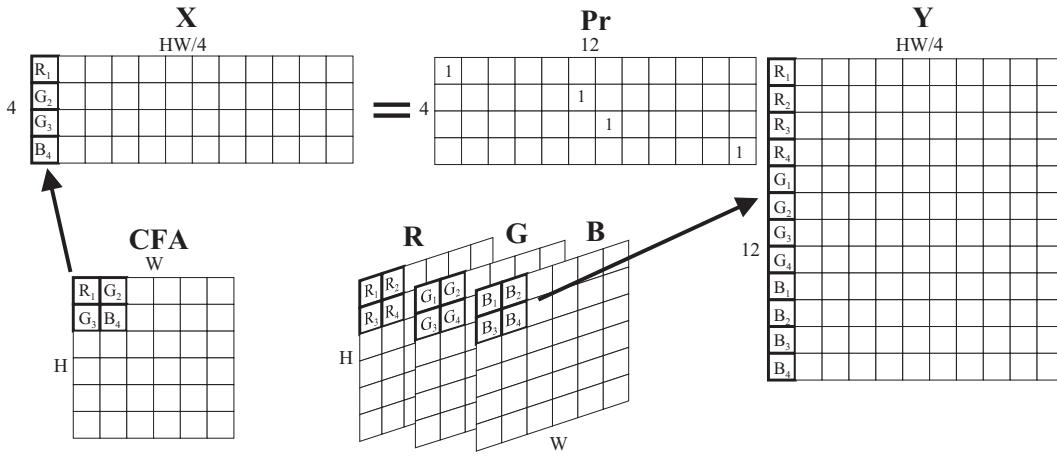
Fig. 2. Illustration that a CFA image $\mathbf{X}$ is constructed from a matrix multiplication between $\mathbf{Pr}$ and the color image $\mathbf{Y}$ if they are represented as column-stacked superpixels.

image $\tilde{\mathbf{Y}}$ from the CFA image $\mathbf{X}$:

$$\tilde{\mathbf{Y}} = \mathbf{D}\mathbf{X} \tag{2}$$

minimizing the mean square error e with the original color image $\mathbf{Y}$:

$$e = E[\|\mathbf{Y} - \tilde{\mathbf{Y}}\|^2] \tag{3}$$

Note that $\mathbf{D}$, in Equation (2), cannot be defined as the inverse of $\mathbf{Pr}$ because $\mathbf{Pr}^T\mathbf{Pr}$ is singular. The classical solution to this equation is the Wiener solution given by:

$$\mathbf{D} = (E[\mathbf{Y}\mathbf{X}^T])(E[(\mathbf{X}\mathbf{X}^T)])^{-1} \tag{4}$$

In order to facilitate the comprehension of the reconstruction, let us consider in a first step the reconstruction of a given superpixel in the color image, considering only the corresponding superpixel in the CFA image. $\mathbf{Y}$ is then a superpixel, a $2 \times 2 \times 3$ matrix, stacked into a 12×1 column vector. The CFA image $\mathbf{X}$ is also a stacked superpixel, but contains only four values, it is a 4×1 column vector. Therefore, the demosaicing matrix $\mathbf{D}$ has to be 12×4, containing the coefficients for reconstructing four pixels with three

4

color components from four pixels with a single color component:

$$
\tilde{\mathbf{Y}} = \mathbf{D} \quad \mathbf{X}
$$

$$
\begin{bmatrix} R_1 \\ \vdots \\ R_4 \\ G_1 \\ \vdots \\ G_4 \\ B_1 \\ \vdots \\ B_4 \end{bmatrix} = \begin{bmatrix} D_{R_1}^1 & \dots & D_{R_1}^4 \\ \vdots & \ddots & \vdots \\ D_{R_4}^1 & \dots & D_{R_4}^4 \\ D_{G_1}^1 & \dots & D_{G_1}^4 \\ \vdots & \ddots & \vdots \\ D_{G_4}^1 & \dots & D_{G_4}^4 \\ D_{B_1}^1 & \dots & D_{B_1}^4 \\ \vdots & \ddots & \vdots \\ D_{B_4}^1 & \dots & D_{B_4}^4 \end{bmatrix} \begin{bmatrix} R_1 \\ G_2 \\ G_3 \\ B_4 \end{bmatrix} \tag{5}
$$

The extension for all the pixels in the image is straightforward, considering the image stacked into a row vector. There are $\frac{H}{2}\frac{W}{2}$ superpixels in the image, assuming that H and W are even. Thus, the vector $\mathbf{X}$ will be of size $4 \times \frac{HW}{4}$, and the corresponding vector $\mathbf{Y}$ will be of size $12 \times \frac{HW}{4}$, as shown in Figure 2. The size of $\mathbf{D}$ remains unchanged. Note that this point differs from [7] and [3] since the images were stacked vertically.

The consideration of a larger kernel for the reconstruction, a neighborhood of superpixels, is obtained by copying the local neighborhoods in vertical direction for matrix $\mathbf{X}$ and by extending $\mathbf{D}$ in horizontal direction. For example for an $n \times n$ neighborhood of superpixels, $\mathbf{Y}$ is still of size $12 \times \frac{HW}{4}$ but $\mathbf{X}$ will be of size $4n^2 \times \frac{HW}{4}$ and $\mathbf{D}$ of size $12 \times 4n^2$. For the remainder of the paper and for ease of notations, stacked images will be implicitly extended vertically with their local neighborhoods.

The matrix product of Equation (2) performs a linear combination of the values of the superpixels neighborhoods weighted by the coefficients of the filter $\mathbf{D}$. In other words, it performs a block shift-invariant convolution between $\mathbf{D}$ and $\mathbf{X}$. The demosaicing matrix $\mathbf{D}$ can be seen as a set of three $4 \times 4n^2$ submatrices $\mathbf{D}_i$, $i \in \{R, G, B\}$ denoting a color plane, which corresponds to the three reconstructing filters of each color plane:

$$
\mathbf{D} = \begin{bmatrix} \mathbf{D}_R \\ \mathbf{D}_G \\ \mathbf{D}_B \end{bmatrix}, \mathbf{D}_i = \begin{bmatrix} D_{i_1}^1 & \dots & D_{i_1}^{4n^2} \\ \vdots & \ddots & \vdots \\ D_{i_4}^1 & \dots & D_{i_4}^{4n^2} \end{bmatrix}
$$

(a) Position 1: G_1 and B_1 filters

(b) Position 2: R_2 and B_2 filters

(c) Position 3: R_3 and B_3 filters
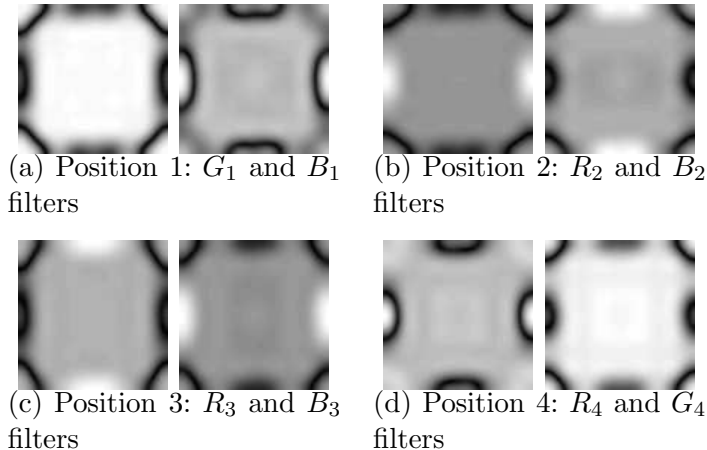

(d) Position 4: R_4 and G_4 filters

Fig. 3. Amplitude spectra of the filters estimated for direct reconstruction at each of the four positions in the Bayer superpixel (filters are normalized for display). The filters differ from one to another in their cutoff frequencies. Their functions are explained by the luminance-chrominance decomposition (section 3).

Each row of the submatrix $\mathbf{D}_i$ corresponds to the reconstructing filter of the component i at one position in the superpixel in $\mathbf{Y}$ from the $4n^2$ elements of the corresponding neighborhood of the superpixel in $\mathbf{X}$.

We can compute matrix $\mathbf{D}$ following Equation (3) over a database of full resolution color images. The use of a database means that we explicitly know $\mathbf{Y}$ and that we simulate the CFA image $\mathbf{X}$ with Equation (1). The computation requires solely the inversion of a matrix of size $4n^2 \times 4n^2$ (n being the size of the neighborhood in superpixels). This result is similar to the one obtained by [3] but computed more directly with our stacked notation. Moreover the fact that the computation is done in the spatial domain rather than in the frequency domain, allows us to control the size of the impulse response (*i.e.* the size of matrix $\mathbf{D}_i$) and to avoid any arbitrary truncation of the impulse response. A similar approach was recently used in [9] through the definition of a spatio-chromatic covariance matrices defined for the four elements of the superpixel.

3 Wiener demosaicing through luminance and chrominance spaces

3.1 Sampling model

Luminance and chrominance coding is a different representation of a color image. $\mathbf{Y}$ can be equally rewritten as the sum of its luminance $\mathbf{\Phi}_c$ and its chrominance $\mathbf{\Psi}_c$ representations: $\mathbf{Y} = \mathbf{\Phi}_c + \mathbf{\Psi}_c$. If we call respectively $\mathbf{P}_c$ and $\mathbf{M}_c$ the luminance and chrominance linear estimators, using the notation of

Figure 2 we can write:
$$\mathbf{Y} = \mathbf{P}_c\mathbf{Y} + \mathbf{M}_c\mathbf{Y} \qquad (6)$$

with $\mathbf{P}_c + \mathbf{M}_c = \mathbf{I}_{12}$ to have the decomposition conservative ($\mathbf{I}_{12}$ the identity matrix of size 12×12). Defining a luminance-chrominance decomposition is equivalent to defining matrices $\mathbf{P}_c$ and $\mathbf{M}_c$. Luminance $\mathbf{\Phi}_c$ is usually defined as a positive weighting of R, G and B values, with weights p_i with $i \in \{R, G, B\}$. We suppose that $\sum_i p_i = 1$, with $p_i \geq 0$, or that luminance is a barycenter of R, G and B values. Moreover, luminance is achromatic, having no chromatic difference information. It thus has a single value per spatial position. It can equivalently be represented as a color vector with three identical intensity values. The chromatic part $\mathbf{\Psi}_c$ of the image called chrominance is then composed by a vector of three components, giving the differences from R, G and B to the luminance image. Since the sum of the p_i equals one, the sum of the chromatic components vanishes at each pixel. Thus, the luminance and chrominance definitions sum up to:

$$\mathbf{P}_c = [\,1\ 1\ 1\,]^T \otimes \underbrace{[\,p_R\ p_G\ p_B\,] \otimes \mathbf{I}_4}_{\mathbf{P}}$$
$$\mathbf{M}_c = \mathbf{I}_{12} - \mathbf{P}_c = (\mathbf{I}_3 - [\,1\ 1\ 1\,]^T \otimes [\,p_R\ p_G\ p_B\,]) \otimes \mathbf{I}_4 \qquad (7)$$

where $\otimes$ is the Kronecker (outer) product. With this notation $\mathbf{P}_c$ is a 12×12 matrix having three times the same submatrix $\mathbf{P}$ in vertical direction (the one detailed on the right-hand side of the equation). The chrominance estimator is a 12×12 matrix. Since the sum of the three coordinates at each spatial position vanishes (linear dependence), the chrominance is intrinsically a two-dimensional signal as usually defined in other luminance and chrominance standards.

Now that we have defined the luminance and chrominance decompostion of a color image, let us see what connection exists with a CFA image. A CFA image can be seen as a grayscale image, meaning a single component per spatial position, like matrix $\mathbf{X}$ in Equation 1, even if this component corresponds alternatively to a spatial multiplexing of different color components. This is illustrated by the fact that $\mathbf{Pr}$ can be decomposed into a sum of three sub-sampling operators $\mathbf{m}_i$ operating on the superpixels of each color plane i as follows:
$$\mathbf{Pr} = [\,1\ 0\ 0\,] \otimes \mathbf{m}_R + [\,0\ 1\ 0\,] \otimes \mathbf{m}_G + [\,0\ 0\ 1\,] \otimes \mathbf{m}_B \qquad (8)$$

with
$$\begin{cases} \mathbf{m}_R = diag([\,1\ 0\ 0\ 0\,]) \\ \mathbf{m}_G = diag([\,0\ 1\ 1\ 0\,]) \\ \mathbf{m}_B = diag([\,0\ 0\ 0\ 1\,]) \end{cases} \qquad (9)$$
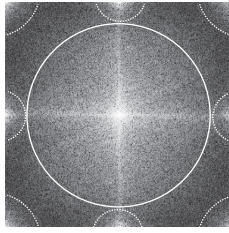
Fig. 4. Average amplitude spectrum of a CFA image computed from all the images in the database: the luminance information is localized at the center while the chrominance information is localized on the corners and on the middle of the borders of the Fourier plane.

where *diag* defines a diagonal matrix filled by the argument vector. Here the "sum" operator acts as the multiplexer since color planes are already subsampled by $\mathbf{m}_i$ functions. A CFA image can equivalently be represented as a color image $\mathbf{X}_c$ in which most of the color values are equal to zero because two third of the chromatic components are missing:

$$\mathbf{X}_c = \mathbf{Pr}^T \mathbf{Pr} \mathbf{Y} = \mathbf{Pr}^T \mathbf{X} \tag{10}$$

The term $\mathbf{Pr}^T\mathbf{Pr}$ denotes a 12×12 subsampling matrix having only four values equal to one, which selects in the color image $\mathbf{Y}$ the pixels corresponding to the CFA image $\mathbf{X}$ and fills the remainder with zeros. As shown on the right-hand side of the equation, $\mathbf{Pr}^T$ operates as a demultiplexer since it isolates each color channel from the mosaic.

We can now rewrite $\mathbf{Pr}$ in terms of luminance and chrominance estimators. Each sampling matrix $\mathbf{m}_i$ can be decomposed as follows: $\mathbf{m}_i = p_i\mathbf{I}_4 + \mathbf{m}_i - p_i\mathbf{I}_4$. Hence, from Equation 8, $\mathbf{Pr}$ can be rewritten:

$$\mathbf{Pr} = [\,p_R\ p_G\ p_B\,] \otimes \mathbf{I}_4 + \mathbf{Pr} - [\,p_R\ p_G\ p_B\,] \otimes \mathbf{I}_4 \tag{11}$$

By identification between Equation 7 and Equation 11 we have:

$$\begin{aligned}
\mathbf{Pr} &= \mathbf{P} + \mathbf{M} \\
&= \mathbf{P} + \mathbf{Pr}(\mathbf{I}_3 \otimes \mathbf{I}_4) - \mathbf{Pr}([\,1\ 1\ 1\,]^T \otimes [\,p_R\ p_G\ p_B\,] \otimes \mathbf{I}_4) \\
&= \mathbf{P} + \mathbf{Pr}\mathbf{M}_c
\end{aligned} \tag{12}$$

which clearly states that the subsampling and projection due to the color filter array does not affect the luminance part of the color image, but is reported on the chrominance part. The projector $\mathbf{Pr}$ is a sum of the luminance estimator $\mathbf{P}$ defined identically as in color images (Equation 7) and a chrominance projector $\mathbf{M}$, which is the projection of the chrominance estimator $\mathbf{M}_c$ defined for color images (Equation 7). It follows that the demultiplexed chrominance in the CFA equals the subsampled chrominance of a color image (i.e.

$\mathbf{Pr}^T\mathbf{M} = \mathbf{Pr}^T\mathbf{Pr}\mathbf{M}_c$). A way to recover the chrominance of a color image from a CFA image is to demultiplex the chrominance part of the CFA (i.e. transforming it into a color image with $\mathbf{Pr}^T$) and to interpolate it.

Moreover, we can constrain parameters p_i to ensure that the modulated chrominance estimator in a CFA image vanishes over a superpixel, and consequently over the whole image. This leads to vanishing the sums of the three diagonals of the matrix $\mathbf{M}$ or in the Bayer CFA:

$$\begin{cases} 1 - 4p_R = 0 \\ 2 - 4p_G = 0 \\ 1 - 4p_B = 0 \end{cases}$$

which gives

$$\left\{ p_R = \frac{1}{4}, \quad p_G = \frac{1}{2}, \quad p_B = \frac{1}{4} \right\}$$

The density spectra of the functions $\mathbf{m}_i$ can be explicitly calculated, as was done in [10]. It appears in the Bayer arrangement that the periodicity of these functions modulates the chrominance to high frequencies on the borders of the spectrum, as shown in Figure 4. The average of the Fourier spectrum of CFA images constructed from the database shows indeed nine regions where the energy is concentrated. The one centered in low frequency corresponds to the non modulated part of the color image in the CFA, the luminance, and the others centered on the corners and on the middle of the sides of the Fourier spectrum correspond to the modulated parts of the color image in the CFA, the chrominance signals.

3.2 Solution derived from the sampling model

Instead of estimating directly the color image from the CFA, as described in the previous section, we can in a first step estimate the luminance $\tilde{\boldsymbol{\Phi}}$ from the CFA:

$$\tilde{\boldsymbol{\Phi}} = \mathbf{H}_{\Phi}\mathbf{X} \tag{13}$$

with $\mathbf{H}_{\Phi}$ being the luminance filter. Once the luminance is estimated, we recover the modulated chrominance as the difference between the CFA image and the luminance $\tilde{\boldsymbol{\Psi}} = (\mathbf{X} - \tilde{\boldsymbol{\Phi}})$. As suggested, we demultiplex the chrominance by multiplying it with $\mathbf{Pr}^T$ before interpolating it to obtain the full chrominance $\tilde{\boldsymbol{\Psi}}_\mathbf{c}$:

$$\tilde{\boldsymbol{\Psi}}_\mathbf{c} = \mathbf{H}_{\Psi}\mathbf{Pr}^T\tilde{\boldsymbol{\Psi}} \tag{14}$$

where $\mathbf{H}_{\Psi}$ is the matrix containing the three chrominance interpolating filters. Finally, the reconstructed color image $\tilde{\mathbf{Y}}$ is the sum of both parts:

$$\tilde{\mathbf{Y}} = \tilde{\boldsymbol{\Phi}}_c + \tilde{\boldsymbol{\Psi}}_c \tag{15}$$

9

Fig. 5. Amplitude spectra of the luminance filter for each position (1, 2, 3 and 4) in the superpixel. At position 2 and 3 (G pixels), luminance can be retrieved with a maximal horizontal and vertical acuity.

where $\tilde{\mathbf{\Phi}}_c = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^T \otimes \tilde{\mathbf{\Phi}}$.

We thus have to train two filters over the database:

- the luminance estimator, calculated from the CFA image $\mathbf{X}$ (which is simulated from the database by setting the appropriate chromatic values to zero) and the luminance $\mathbf{\Phi}$ (which is also computed from the database):

$$\mathbf{H}_\Phi = (E[\mathbf{\Phi}\mathbf{X}^T])(E[(\mathbf{X}\mathbf{X}^T)])^{-1} \tag{16}$$

- and the chrominance interpolator, calculated from the chrominance $\mathbf{\Psi_c}$ and the subsampled chrominance $\mathbf{\Psi}$ (both computed from the database):

$$\mathbf{H}_\Psi = (E[\mathbf{\Psi_c}(\mathbf{Pr}^T\mathbf{\Psi})^T])(E[(\mathbf{Pr}^T\mathbf{\Psi})(\mathbf{Pr}^T\mathbf{\Psi})^T)])^{-1} \tag{17}$$

with $\mathbf{\Phi} = \mathbf{PY}$, $\mathbf{\Psi}_c = \mathbf{M}_c\mathbf{Y}$ and $\mathbf{\Psi} = \mathbf{X} - \mathbf{\Phi}$. The great advantage of this decomposition is that the chrominance has a narrow bandwidth with respect to the Nyquist frequency of each demultiplexed plane. It requires thus only small order filters for interpolation. At the opposite, the luminance estimator needs to have high gradients at the frequencies located on the border between luminance and modulated chrominance. It requires thus a high order filter for estimation (typically 7×7 or 9×9), but at least this estimation is performed only once. This property makes the algorithm computationally much more efficient than the RGB algorithm presented in the previous section.

4   Results and discussion

In this section, after a quick state of the art in demosaicing, we describe the results of an implementation of the proposed method using an image database and compare it to existing algorithms in term of quality and performance.

(a) Position 1: G_1 and B_1 filters

(b) Position 2: R_2 and B_2 filters

(c) Position 3: R_3 and B_3 filters
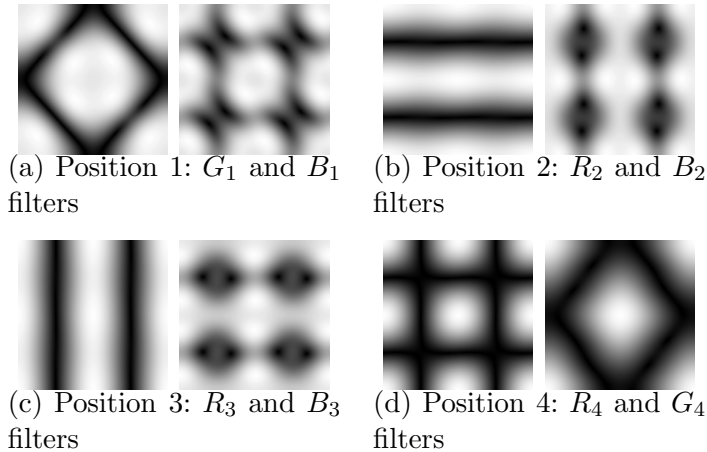

(d) Position 4: R_4 and G_4 filters

Fig. 6. Amplitude spectra of the chrominance filters at each position in the super-pixel. Since chrominance is a low-pass signal, low-order filters (3×3) can be used.

4.1 State of the art

There are many ways to define a useful constraint set for improving the reconstruction of the image. Cok [11] proposed the use of constant hue hypothesis, interpolating hue (ratio of two colors) instead of the color channel because hue is rather constant on object surface. The constant hue hypothesis was also used by Adams [4] to design convolution filters. More recently, Lukac [12] proposed a normalized (scaled and shifted) color ratio that improved color ratios interpolation. Many authors (see [13] for a review) have proposed directional interpolation to account edges where most visible artifacts appear in the reconstructed image [14–18]. These methods result in a weighted bilinear interpolation where the weights depend on the local content of the image. A variant edge-sensing method consists in interpolating separately in horizontal and vertical directions and chosing the prefered directions according a criteria based on gradients [14,19,20]. Other authors proposed an iterative process or post-processing process regularizing the reconstructed image, possibly combined to an edge-sensitive interpolation [13,21,22,18,23–26]. In [27] the authors used the high frequency pattern of the green channel, which has a better resolution in the Bayer CFA, to restrict the set of solutions by artificially copying this high frequency pattern onto the red and blue planes. In an extended version [28], two constrained sets were defined, one garanteed that the high frequency components were those corresponding to the green channel, and the second one ensured that existing pixels in the CFA kept the same values in the reconstructed image.

Turning to linear methods, which recently become attractive due to their simplicity, effectiveness and quality, Crane *et al.* [29] proposed to use the constant hue hypothesis [11] to design convolution filters for the interpolation. Pei *et al* [30] interpolated bilinearly the color difference R minus G and B minus G.

11

Malvar *et al.* [31] proposed a linear method based on the assumption that edges would have much stronger "luminance" than "chrominance" components. This assumption was used by correcting the result of a bilinear interpolation with a coefficient expressing the local "luminance" variations. In Alleysson *et al.* [10], it was shown that CFA sampling modulates luminance and chrominance in the Fourier domain. This allows estimating luminance and chrominance by frequency selection and results in cost effective, linear, space-invariant convolution filters for demosaicing. Based on this model, Lian *et al.* [26] proposed using a space-variant filter to improve luminance estimation according to the structure of Bayer CFA. Both [26,32] include an adaptive process in the luminance estimation, resulting in a high quality reconstruction.

There is a more explicit way to construct the constrained set using the estimated statistics of color images. Muresan *et al.* [33] built a metric based on exemplars, which defined a restricted set of solutions for the interpolation. In [34], the authors proposed to estimate the statistics of the image with the existing pixels and to design image-specific Wiener filters. In their papers, Taubman [7] and Trussell *et al.* [3] used an image formation model, where the source was a hyperspectral image. They used a minimum mean square error approach with prior probability distributions on the image formation to design filters for demosaicing and deblurring the hyperspectral image acquired through a CFA-based camera. In Portilla *et al.* [9] the filters were estimated from an image database and it is shown that LLMSE demosaicing can operate denoising and deblurring. In Hirakawa *et al.* [35], a TLS denoising is designed conjointly with demosaicing and operated well on real noisy CMOS images.

4.2   Implementation and performance analysis

We chose to implement the Wiener methods of sections 2 and 3 by training the filters on an image database. The main assumption is that all the images of the database have been acquired through the same device, so that they share common statistics. This assumption allows us defining a unique optimal linear demosaicing filter for the image database. However it limits its application to a specific camera because the spectral and the optical characteristics of the camera influence the designed filters. One solution for designing the filters consists, as Taubman proposed [7], in estimating the spectral and optical behaviors of the camera and including it in the filter. The use of a color image database having the spectral and optical behaviour as the designed camera is an alternative solution. Moreover, the use of a database allows to combine in a single operation different color processing steps employed in the image pipeline, because the filter will follow the behavior of the database. An example of sharpening joint to demosaicing was given by Portilla *et al.* [9].

Twenty-four color images from the Kodak database[3] (768×512 pixels) were used for the simulation. We computed the mean Peak Signal-to-Noise Ratio and the S-CIELAB metric ΔE (assuming a 72 dpi monitor, 18 in. of distance) between the original color images and the reconstructed images using the "leave-one-out" method [36]. This method is widely used in the field of data analysis when the number of available elements for training and testing is restricted. If the database contains N elements, the "leave-one-out" method consists in training the algorithm (computing $\mathbf{D}$) over (N-1) elements and testing it (computing $\hat{\mathbf{Y}}$) only over the left out element. This operation is repeated for each element of the database, and the results are averaged over the whole database. It permits avoiding the obvious bias when a tested image was used in the training set, while maximising the number of elements in the training set.

The simulation over the image database gave an average PSNR of 39.20dB for the Wiener demosaicing algorithm (Table 1, row labeled [9]) for kernels of size 9×9 pixels, and 38.63dB for kernels of size 7×7. The resulting two-dimensional filters reconstructed from the stacked filters (the rows of D) for each of the four positions in the superpixel are shown in Figure 3. Obviously, only two filters are represented at each position of the superpixel instead of three; the third one estimating a color value from an existing one, being the identity. We see that the filters differ from one to another in their high frequency characteristics.

Concerning the luminance/chrominance algorithm, we estimated one luminance filter of size 9×9 pixels (Figure 5) and one of size 7×7, and chrominance filters of size 3×3 pixels (Figure 6). We obtained average PSNR's of respectively 39.16dB and 38.67dB with the 9×9 filter and with the 7×7 filter. As expected by the model, the luminance filters are large low-pass filters cutting the areas where the chrominance is modulated. Interestingly the luminance has a full vertical and horizontal resolution at positions 2 and 3 in the superpixel. This property was demonstrated and exploited in [26] by considering the fact that the sum of the two high vertical frequency and high horizontal frequency carriers vanishes on G pixels. Moreover, as [26] remarked in his paper, there is less aliasing with modulated chrominance at vertical and horizontal frequencies than with chrominance modulated at diagonal frequencies. This allows using a smaller order filter on G pixels. By consequence a 5×5 kernel was designed for usage on the G pixels, without any significative loss of quality (0.05 dB). The results for the image database appear in Table 1, row called *Proposed*. They are very close to those found with the algorithm described in previous section and visual artefacts are the same in both cases (Figures 7 and 8).
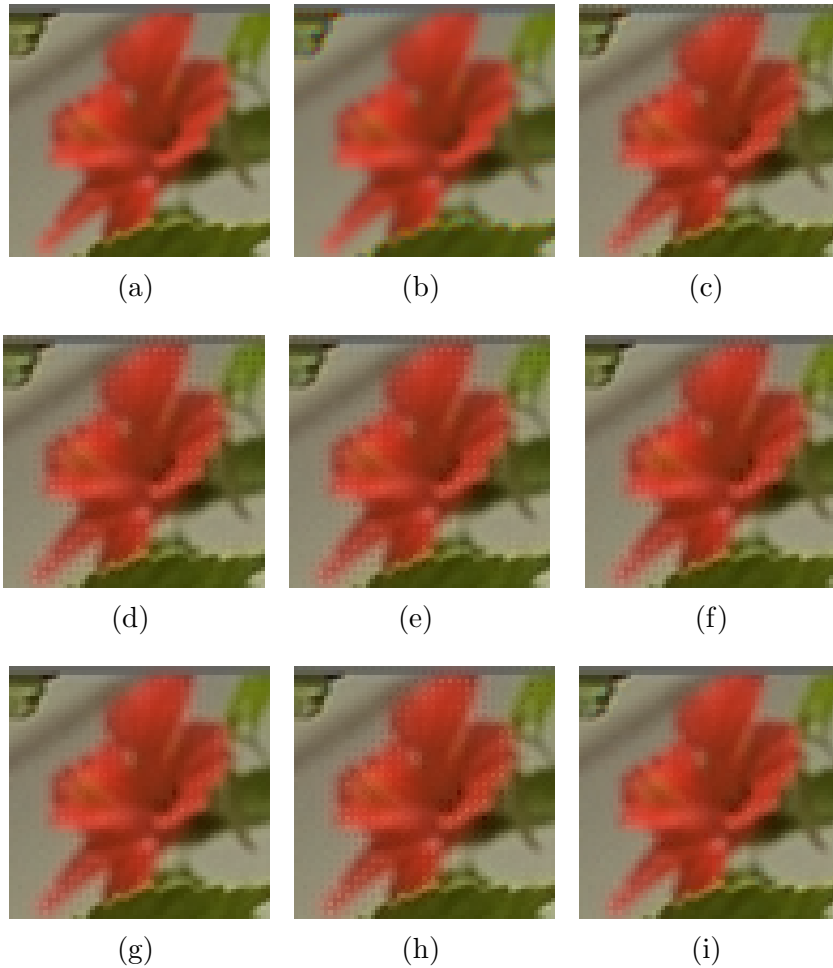

Fig. 7. Example illustrating the zipper effect: (a) original image, and reconstruction using (b) bilinear method, (c) [30], (d) [9], (e) proposed method, (f) [32], (g) [26], (h) [25] and (i) [19].

Visually, both algorithms may suffer from zipper noise near edges (Figure 7) and false colors in high frequency areas (Figure 8). Zipper noise appear when chrominance has strong variations. The modulated chrominance overflows then on luminance and is actually interpreted as high frequency pattern of luminance. On the oppposite, false colors appear when luminance is of too high frequency. Luminance then overflows on the modulated chrominance and is demodulated into low frequency as a chrominance signal.

In Table 1 are also represented the results for the bilinear method and for methods of [30,31,10,28,26,32,25,19]. The results of some of these algorithms are also represented in Figures 7 and 8. [28] was tested with 3 and 8 iterations, [25] with one and two thresholds. Among the tested algorithms, the one reducing both zipper noise and false colors most is [19].
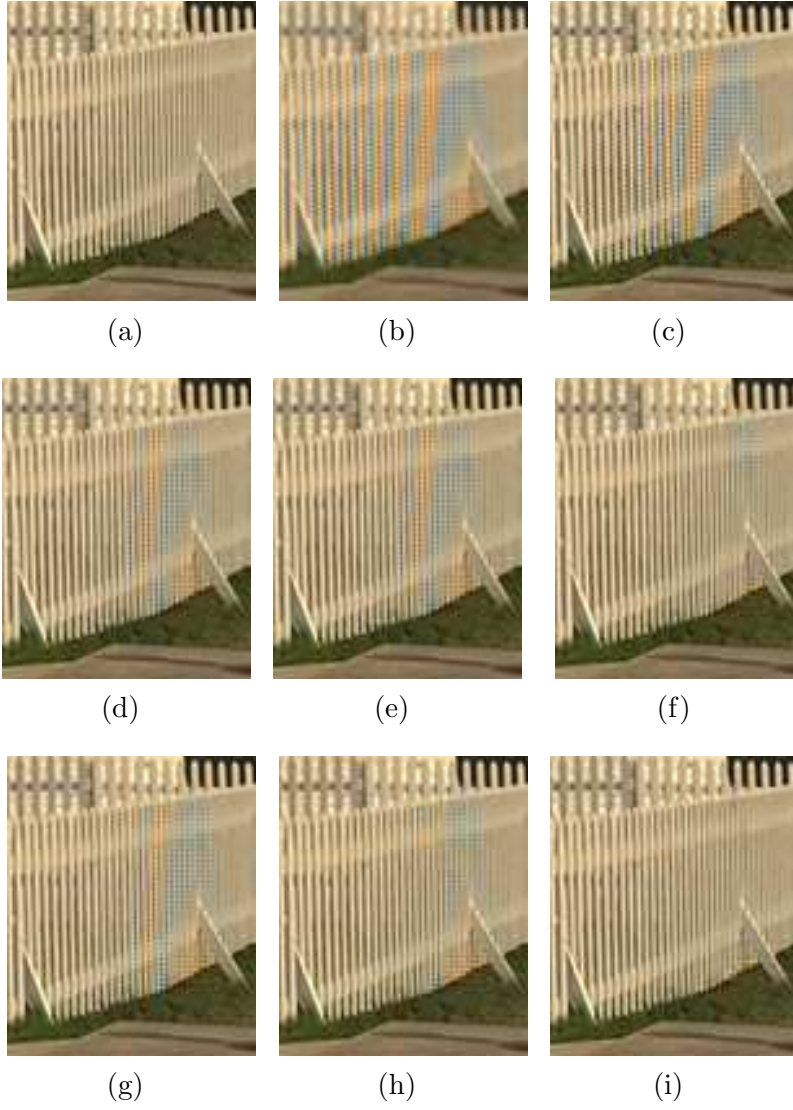
Fig. 8. Example illustrating the false color artefact, in the same order as in Figure 7.

4.3 Link between RGB algorithm and luminance-chrominance algorithm

In the RGB algorithm, each color plane is retrieved by picking up the optimal combination of pixels in the CFA image using the filter $\mathbf{D}$, while in the luminance/chrominance algorithm the image is reconstructed from the sum of luminance and chrominance. It can be easily shown that these two methods are formally equivalent. Combining Equations (3) and (15) gives:

$$\mathbf{D} = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^{T} \otimes \mathbf{H}_{\phi} + \mathbf{H}_{\psi}\mathbf{P_{r}}^{T}\left(\mathbf{I_{4}} - \mathbf{H}_{\phi}\right) \tag{18}$$

method	R	G	B	$\Delta E_{S-CIELAB}$	Efficiency
bilinear	29.22 (±3.32)	33,04 (±3.25)	29.27 (±3.33)	1.79 (±0.81)	$6HW$
[30]	37.21 (±3.12)	38.87 (±2.99)	36.06 (±2.99)	1.12 (±0.44)	$12HW$
[31]	35.36 (±3.34)	38.87 (±2.99)	34.15 (±3.12)	1.25 (±0.53)	$21HW$
[10]	37.83 (±2.44)	40.74 (±2.26)	36.48 (±2.38)	1.10 (±0.34)	$77HW$
[28] 3 iter.	38.40 (±2.73)	41.37 (±2.40)	37.46 (±2.59)	0.99 (±0.35)	$405HW$
[28] 8 iter.	39.29 (±2.54)	41.37 (±2.40)	37.81 (±2.47)	0.96 (±0.33)	$885HW$
[26]	38.77 (±2.59)	42.12 (±2.79)	38.62 (±2.88)	0.82 (±0.31)	$63HW$
[32]	38.81 (±2.50)	42.82 (±2.50)	38.62 (±2.69)	0.83 (±0.30)	$2274HW$ [a]
[25] 1 thr.	38.37 (±2.45)	41.77 (±2.40)	38.44 (±2.67)	0.93 (±0.33)	$77HW$
[25] 2 thr.	38.07 (±2.45)	41.37 (±2.40)	38.12 (±2.65)	0.96 (±0.35)	$117HW$
[19]	38.02 (±3.24)	39.59 (±3.21)	36.76 (±3.06)	0.90 (±0.37)	$161HW$
[9], 7x7	38.15 (±2.55)	40.88 (±2.41)	36.86 (±2.55)	1.15 (±0.34)	$98HW$
[9] 9x9	38.87 (±2.56)	41.43 (±2.42)	37.29 (±2.58)	1.00 (±0.34)	$162HW$
Proposed, 7x7	38.40 (±2.49)	40.88 (±2.37)	36.74 (±2.47)	1.18 (±0.34)	$47HW$
Proposed, 9x9	38.92 (±2.52)	41.34 (±2.36)	37.21 (±2.51)	1.01 (±0.33)	$63HW$

[a] this value is not significative since the author of [32] has not optimized his algorithm.
Table 1
Mean PSNR values for R, G and B planes (dB); mean ΔE in the S-CIELAB computed over the database; and efficiency in number of cycles for an image of size HxW, for several alogrithms.

and the definition of the luminance estimator (Equation (7)) yields to:

$$\begin{cases} \mathbf{H}_\phi = \sum_i p_i \mathbf{D}_i \\ \mathbf{H}_{\psi i} = (\mathbf{D}_i - \mathbf{H}_\phi)(\mathbf{I}_4 - \mathbf{H}_\phi)^{-1} \end{cases} \qquad (19)$$

This explains why the obtained PSNR values for both algorithms (Table 1) are so close. Moreover we can now interpret the shapes of the filters encountered in Figure 3. All filters have large low-pass components, which correspond to the luminance filter, and the corners and the middle of the sides are present or not, following the position in the superpixel in order to demodulate the chrominance part.

The performances of the various algorithms are estimated in terms of number of clock cycles required for processing an image of size HxW. Since a convolution can be efficiently implemented on a Digital Signal Processor with the MAC instruction (Multiplier-Accumulator), we consider that the multiplication by one coefficient and the addition of the result is performed in one cycle (it is true for most DSPs). Other instructions (absolute values, comparisons, divisions...) are also counted as one cycle, although they may take more cycles following the DSP model. The instructions of data transfer and data reading/writing are not taken into account. The values are reported in Table 1 at the *Efficiency* column. For the Wiener RGB method, we have two convolutions with 9×9 filters, yielding to $162HW$ cycles. For the luminance-chrominance method, a 9×9 or 7×7 luminance filter is used for R/B pixels and a 5×5 filter is used for G pixels (53 or 37 cycles), and then two 3×3 chrominance filters (6 cycles, as bilinear filtering: 4 operations at G pixels and 8 operations at R or B pixels). Taking account the substraction of the luminance from the mosaic image (1 cycle) and its addition to each color plane (3 cycles), this leads to a complexity of $63HW$ cycles when a 9×9 filter is used at R/B pixels and $47HW$ cycles when a 7×7 one is used, almost one third of that of the direct RGB algorithm [9] for the same quality. Note that in [8], Hel-Or proposed a method to obtain unique filters for each pixel for block shift-invariant algorithms in order to apply efficiently the algorithm using a unique block-shift invariant convolution. At the opposite, we see that for the method described in the present paper, decomposing the global impulse responses into two steps (luminance filter and then chrominance filters) is algorithmically more efficient.

5   Conclusion

We presented a practical implementation of a linear Wiener demosaicing using a novel stacked superpixel notation which was tested on an image database with the Bayer color filter array. This algorithm picks up the optimal linear combinations of pixels in the CFA to retrieve the color image using three high-order filters, one per color channel. By decomposing the single-sensor image into luminance and chrominance, the algorithm can be implemented in a much more efficient manner without any loss of quality. Indeed, the chrominance signal being a low-pass signal, low-order filters may be used for its interpolation. Only the luminance filters has to be high order. This decomposition results in the increase of efficiency in the implementation of linear demosaicing algorithms.

Fig. 9. Additional cropped images reconstructed with the present method.

Non linear methods, such as directional interpolation, result in a reconstruction with less artefacts, but the spectral considerations of luminance and chrominance in CFA images may be helpful for the understanding of color mosaic images and, more generally, for the understanding of color imaging.

Acknowledment

References

[1] R. Lukac, B. Smolka, K. Martin, K. N. Plataniotis, A. N. Venetsanopoulos, Vector filtering for color imaging, IEEE Signal Processing Magazine 22 (1) (2005) 74–86.

[2] R. Lukac, K. N. Plataniotis, Color filter arrays: Design and performance analysis, IEEE Transactions on Consumer Electronics 51 (4) (2005) 1260–1267.

[3] H. J. Trussell, R. E. Hartwig, Mathematics for demosaicking, IEEE Transactions on Image Processing 11 (4) (2002) 485–492.

[4] J. E. Adams, Design of practical color filter array interpolation algorithms for digital cameras, in: SPIE, Vol. 3028, 1997, pp. 117–125.

[5] T. Sakamoto, C. Nakanishi, T. Hase, Software pixel interpolation for digital still cameras suitable for a 32-bit mcu, IEEE Transactions on Consumer Electronics 44 (4) (1998) 1342–1352.

[6] B. Bayer, Color imaging array, US patent 3,971,065, to Eastman Kodak Company (1976).

[7] D. Taubman, Generalized Wiener reconstruction of images from colour sensor data using a scale invariant prior, in: IEEE International Conference on Image Processing, Vol. 3, 2000, pp. 801–804.

[8] Y. Hel-Or, The impulse responses of block shift-invariant systems and their use for demosaicing algorithms, in: IEEE International Conference on Image Processing, Vol. 2, 2005, pp. 1006–1009.

[9] J. Portilla, D. Otaduy, C. Dorronsoro, Low-complexity linear demosaicing using joint spatial-chromatic image statistics, in: IEEE International Conference on Image Processing, Vol. 1, Genoa, Italy, 2005, pp. 61–64.

[10] D. Alleysson, S. Süsstrunk, J. Hérault, Linear color demosaicing inspired by the human visual system, IEEE Transactions on Image Processing 14 (2005) 439–449.

[11] D. R. Cok, Signal processing method and apparatus for producing interpolated chrominance values in a sampled color image signal, US patent 4,642,678, to Eastman Kodak Company (1987).

[12] R. Lukac, K. N. Plataniotis, Normalized color-ratio modeling for CFA interpolation, IEEE Transactions on Consumer Electronics 50 (2) (2004) 737–745.

[13] B. K. Gunturk, J. Glotzbach, Y. Altunbazak, R. W. Schafer, R. M. Mersereau, Demosaicking: Color filter array interpolation in single-chip digital cameras, IEEE Signal Processing Mag. 22 (1) (2005) 44–54.

[14] J. F. Hamilton, J. Adams, Adaptive color plane interpolation in single sensor color electronic camera, US Patent 5,629,734, to Eastman Kodak Company (May 1997).

[15] L. Chang, Y.-P. Tan, Effective use of spatial and spectral correlations for color filter array demosaicking, IEEE Transactions on Consumer Electronics 50 (1) (2004) 355–365.

[16] R. Lukac, K. N. Plataniotis, D. Hatzinakos, Color image zooming on the Bayer pattern, IEEE Transactions on Circuits and Systems for Video Technology 15 (11) (2005) 1475–1492.

[17] X. Wu, N. Zhang, Primary-consistent soft-decision color demosaicking for digital cameras, IEEE Transactions on Image Processing 13 (9) (2004) 1263–1274.

[18] W. Lu, Y.-P. Tan, Color filter array demosaicking: New method and performance measures, IEEE Transactions on Image Processing 12 (10) (2003) 1194–1210.

[19] K. Hirakawa, T. W. Parks, Adaptive homogeneity-directed demosaicing algorithm, IEEE Transactions on Image Processing 14 (3) (2005) 360–369.

[20] L. Zhang, X. Wu, Color demosaicking via directional linear minimum mean square-error estimation, IEEE Transactions on Image Processing 14 (12) (2005) 2167–2178.

[21] R. Kimmel, Demosaicing: Image reconstruction from color samples, IEEE Transaction On Image Processing 8 (1999) 1221–1228.

[22] R. Lukac, K. Plataniotis, D. Hatzinakos, M. Aleksic, A novel cost effective demosaicing approach, IEEE Transactions on Consumer Electronics 50 (1) (2004) 256– 261.

[23] R. Lukac, K. Martin, K. N. Plataniotis, Demosaicked image post-processing using local color ratios, IEEE Transactions on Circuits and Systems for Video Technology 14 (6) (2004) 914–920.

[24] R. Lukac, K. N. Plataniotis, A robust, cost-effective postprocessor for enhancing demosaicked camera images, Real-Time Imaging, Special Issue on Spectral Imaging II 11 (2) (2005) 139–150.

[25] X. Li, Demosaicing by successive approximations, IEEE Transactions on Image Processing 14 (3) (2005) 370–379.

[26] N. Lian, L. Chang, Y.-P. Tan, Improved color filter array demosaicking by accurate luminance estimation, in: IEEE International Conference on Image Processing, Vol. 1, 2005, pp. 41–44.

[27] J. Glotzbach, R. Schafer, K. Illgner, A method of color filter array interpolation with alias cancellation properties, in: IEEE International Conference on Image Processing, Thessaloniki, Greece, 2001, pp. 141–144.

[28] B. Gunturk, Y. Altunbasak, R. Mersereau, Color plane interpolation using alternating projections, IEEE Transactions on Image Processing 11 (9) (2002) 997–1013.

[29] H. D. Crane, J. D. Peter, E. Martinez-Uriegas, Method and apparatus for decoding spatiochromatically multiplexed color image using predetermined coefficients, US patent 5,901,242, to SRI International (1999).

[30] S.-C. Pei, I.-K. Tam, Effective color interpolation in ccd color filter arrays using signal correlation, IEEE Transactions on Circuits and Systems for Video Technology 13 (6).

[31] H. Malvar, L.-W. He, R. Cutler, High-quality linear interpolation for demosaicing of Bayer-patterned color images, in: IEEE International Conference on Acoustics, Speech and Signal Processing, Vol. 3, Montréal, Canada, 2004, pp. 485–488.

[32] E. Dubois, Frequency-domain methods for demosaicking of Bayer-sampled color images, IEEE Signal Processing Letters 12 (2005) 847–850.

[33] D. Muresan, T. Parks, Demosaicing using optimal recovery, IEEE Transactions on Image Processing 14 (2) (2005) 267–278.

[34] X. Li, M. Orchard, New edge-directed interpolation, IEEE Transactions on Image Processing 10 (10) (2001) 1521–1527.

[35] K. Hirakawa, T. W. Parks, Joint demosaicing and denoising, IEEE Transactions on Image Processing 15 (8) (2006) 2146–2157.

[36] K. Fukunaga, D. M. Hummels, Leave-one-out procedures for nonparametric error estimates, IEEE Transactions on Pattern Analysis and Machine Intelligence 11 (4) (1989) 421–423.