Comparison of color demosaicing methods

Olivier Losson, Ludovic Macaire, Yanqin Yang

Comparison of color demosaicing methods

O. Losson[a,*], L. Macaire[a], Y. Yang[a]

[a] *Laboratoire LAGIS UMR CNRS 8146 – Bâtiment P2*
Université Lille1 – Sciences et Technologies, 59655 Villeneuve d'Ascq Cedex, France

Keywords: Demosaicing, Color image, Quality evaluation, Comparison criteria

1. Introduction

Today, the majority of color cameras are equipped with a single CCD (Charge-Coupled Device) sensor. The surface of such a sensor is covered by a color filter array (CFA), which consists in a mosaic of spectrally selective filters, so that each CCD element samples only one of the three color components Red (*R*), Green (*G*) or Blue (*B*). The Bayer CFA is the most widely used one to provide the CFA image where each pixel is characterized by only one single color component. To estimate the color (*R*,*G*,*B*) of each pixel in a true color image, one has to determine the values of the two missing color components at each pixel in the CFA image. This process is commonly referred to as CFA demosaicing, and its result as the demosaiced image. In this paper, we propose to compare the performances reached by the demosaicing methods thanks to specific quality criteria.

An introduction to the demosaicing issue is given in section 2. Besides explaining why this process is required, we propose a general formalism for it. Then, two basic schemes are presented, from which are derived the main principles that should be fulfilled in demosaicing.

In section 3, we detail the recently published demosaicing schemes which are regrouped into two main groups : the spatial methods which analyze the image plane and the methods which examines the frequency domain. The spatial methods exploit assumptions about either spatial or spectral correlation between colors of neighbors. The frequency-selection methods apply specific filters on the CFA image to retrieve the color image.

Since these methods intend to produce "perceptually satisfying" demosaiced images, the most widely used evaluation criteria detailed in section 4 are based on the fidelity to the original images. Generally, the Mean Square Error (*MSE*) and the Peak Signal-to-Noise Ratio (*PSNR*) are used to measure the fidelity between the demosaiced image and the original one. The *PSNR* criterion cannot distinguish the case when there are

*Corresponding author
Email addresses: olivier.losson@univ-lille1.fr (O. Losson), ludovic.macaire@univ-lille1.fr (L. Macaire), yanqin.yang@etudiant.univ-lille1.fr (Y. Yang)

a high number of pixels with slight estimation errors from the case when only a few pixels have been interpolated with severe demosaicing artifacts. However, the latter case would more significantly affect the result quality of a low-level analysis applied to the estimated image. Therefore, we propose new criteria especially designed to determine the most effective demosaicing method for further feature extraction.

The performances of the demosaicing methods are compared in section 5 thanks to the presented measurements. For this purpose, the demosaicing schemes are applied to twelve images of the benchmark Kodak database.

2. Color Demosaicing

Digital images or videos are currently a preeminent medium in environment perception. They are today almost always captured directly by a digital (still) camera, rather than digitized from a video signal provided by an analog camera as they used to be several years ago. Acquisition techniques of *color* images in particular have involved much research work and undergone many changes. Despite major advancements, mass-market color cameras still often use a single sensor and require subsequent processing to deliver color images. This procedure, named *demosaicing*, is the key point of our study and is introduced in the present section. The demosaicing issue is first presented in detail, and a formalism is introduced for it.

2.1. Introduction to the Demosaicing Issue

The demosaicing issue is here introduced from technological considerations. Two main types of color digital cameras are found on the market, depending on whether they embed three sensors or a single one. Usually known as *mono-CCD* cameras, the latter are equipped with spectrally-sensitive filters arranged according to a particular pattern. From such color filter arrays (CFA), an intermediate gray-scale image is formed, which then has to be *demosaiced* into a true color image.

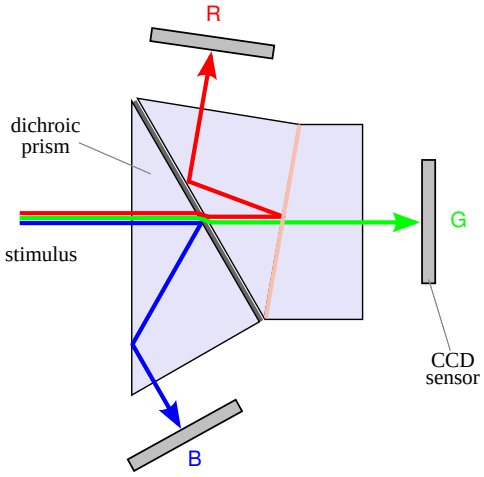
In the first subsection are compared the major implementations of three-CCD and mono-CCD technologies. Then are presented the main types of colors filter arrays released by the various manufacturers. Proposed by Bayer from Kodak in 1976, the most widespread CFA is considered in the following, not only to formalize demosaicing but also to introduce a pioneer method using bilinear interpolation. This basic scheme generates many color artifacts, which are analyzed to derive two main demosaicing rules. Spectral correlation is one of them, and will be detailed in the last subsection. The second one, spatial correlation, is at the heart of edge-adaptive demosaicing methods, that will be presented in the next section.

2.1.1. Mono-CCD vs. Three-CCD Color Cameras

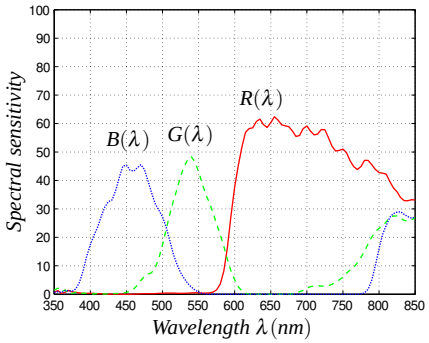
Digital area scan cameras are devices able to convert color stimuli from the observed scene into a color digital image (or image sequence) thanks to photosensors. Such an output image is spatially digitized, being formed of picture elements (*pixels*). With each pixel is generally associated a single photosensor element, which captures the incident light intensity of the color stimulus.

A digital color image $\mathbf{I}$ can be represented as a matrix of pixels, each of them being denoted as $P(x,y)$, where x and y are the spatial coordinates of pixel P within the image plane of size $X \times Y$, hence $(x,y) \in \mathbb{N}^2$ and $0 \leqslant x \leqslant X-1$, $0 \leqslant y \leqslant Y-1$. With each pixel P is associated a color point, denoted as $\mathbf{I}(x,y)$ or $\mathbf{I}_{x,y}$. This color point is defined in the *RGB* three-dimensional color space by its three coordinates $I^k_{x,y}$, $k \in \{R,G,B\}$, which represent the levels of the trichromatic components of the corresponding color stimulus.

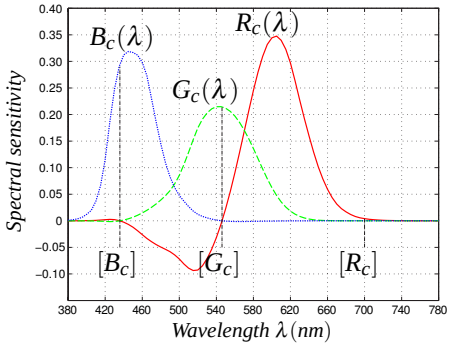
The color image $\mathbf{I}$ may also be split into three *component planes* or *images* I^k, $k \in \{R,G,B\}$. In each component image I^k, the pixel P is characterized by level $I^k(P)$ for the single color component k. Thus, three component images I^R, I^G and I^B, must be acquired in order to form any digital color image.



(a) Beam splitting by a trichroic prism assembly.



(b) Relative spectral sensitivity of the Kodak KLI-2113 sensor.

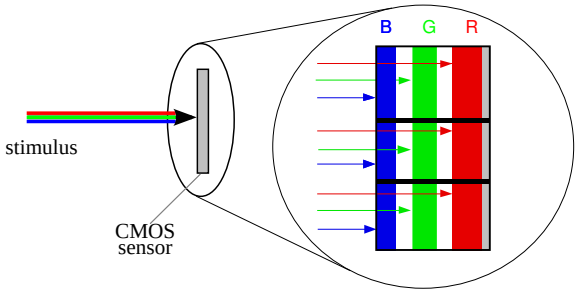


(c) CIE 1931 RGB color matching functions. $[R_c]$, $[G_c]$ and $[B_c]$ are the monochromatic primary colors.

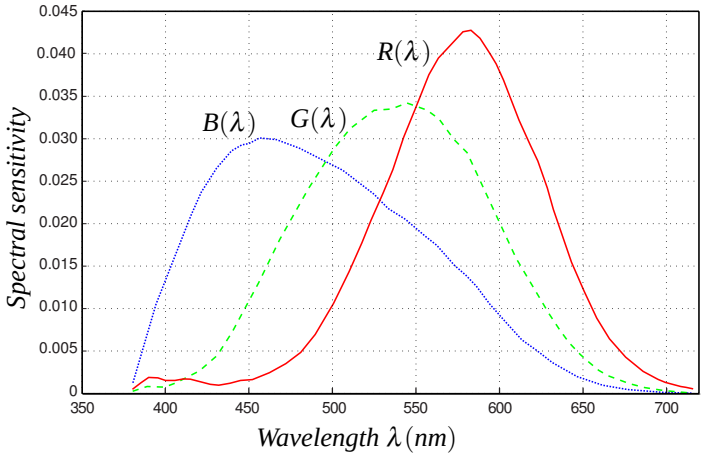
FIG. 1: Three-CCD technology.

The two main technology families available for the design of digital camera photosensors are CCD (*Charge-Coupled Device*) and CMOS (*Complementary Metal-Oxide Semiconductor*) technologies, the former being the most widespread one today. The CCD technology uses the photoelectric effect of the silicon substrate, while CMOS is based on a photodetector and an active amplifier. Both photosensors overall convert the intensity of the light reaching each pixel into a proportional voltage. Additional circuits then converts this analog voltage signal into digital data. For illustration and explanation purposes, the following text relates to the CCD technology.

The various digital color cameras available on the market may also be distinguished according to whether they incorporate only a single sensor or three. In accordance with the trichromatic theory, three-CCD technology incorporates three CCD sensors, each one being dedicated to a specific primary color. In most devices, the color stimulus from the observed scene is split onto the three sensors by means of a trichroic



(a) Wavelength absorption within the Foveon X3 sensor.



(b) Relative spectral sensitivity of the Foveon X3 sensor endowed with an infrared filter (Lyon and Hubel, 2002).

FIG. 2: Foveon X3 technology.

prism assembly, made of two dichroic prisms (see figure 1a)(Lyon, 2000). Alternately, the incident beam may be dispatched on three sensors, each one being covered with a spectrally selective filter. The three component images I^R, I^G and I^B are simultaneously acquired by the three CCD sensors, and their combination leads to the final color image. Each digital three-CCD camera is characterized by its own spectral sensitivity functions $R(\lambda)$, $G(\lambda)$ and $B(\lambda)$ (see figure 1b for an example), which differ from the CIE color matching functions of the standard observer (see figure 1b).

Since 2005, Foveon Inc. has been developing the X3 sensor, which uses a multilayer CMOS technology. This new sensor is based on three superimposed layers of photosites embedded in a silicon substrate. It takes advantage of the fact that lights of different wavelengths penetrate silicon to different depths (see figure 2a)(Lyon and Hubel, 2002). Each layer hence captures one of the three primary colors, namely blue, green and red, in the light incidence order. The three photosites associated with each pixel

thus provide signals from which the three component values are derived. Any camera equipped with this sensor is able to form a true color image from three full component images, as do three-CCD-based cameras. This sensor has been first used commercially in 2007 within the Sigma SD14 digital still camera. According to its manufacturer, its spectral sensitivity (see figure 2b) better fits with the CIE color matching functions than those of three-CCD cameras, providing images that are more consistent with human perception.

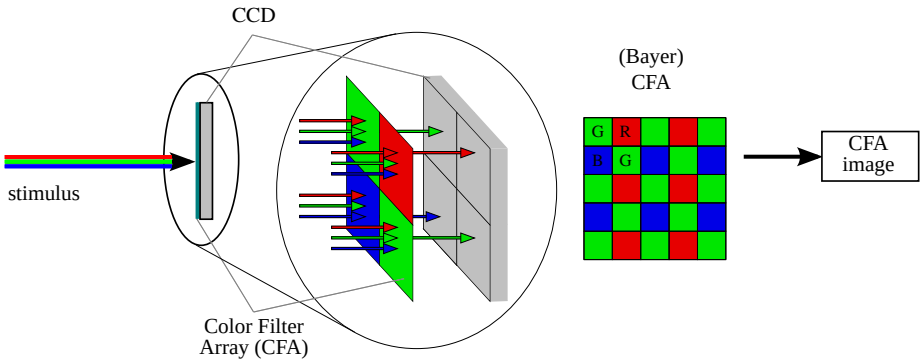
Although three-CCD and Foveon technologies yield high quality images, the manufacturing costs of the sensor itself and of the optical device are high. As a consequence, cameras equipped with such sensors have not been so far affordable to everyone, nor widely distributed.

In order to overcome these cost constraints, a technology using a single sensor has been developed. The solution suggested by Bayer from the Kodak company in 1976 (Bayer, 1976) is still the most widely used in commercial digital cameras today. It uses a CCD or CMOS sensor covered by a filter (*Color Filter Array*, or CFA) designed as a mosaic of spectrally selective color filters, each of them being sensitive to a specific wavelength range. At each element of the CCD sensor, only one out of the three color components is sampled, Red (R), Green (G) or Blue (B) (see figure 3a). Consequently, only one color component is available at each pixel of the image provided by the CCD charge transfer circuitry. This image if often related to as the *raw* image, but *CFA image* is preferred hereafter in our specific context. In order to obtain a color image from the latter, two missing levels must be estimated at each pixel thanks to a *demosaicing* algorithm (sometimes spelled *demosaicking*).

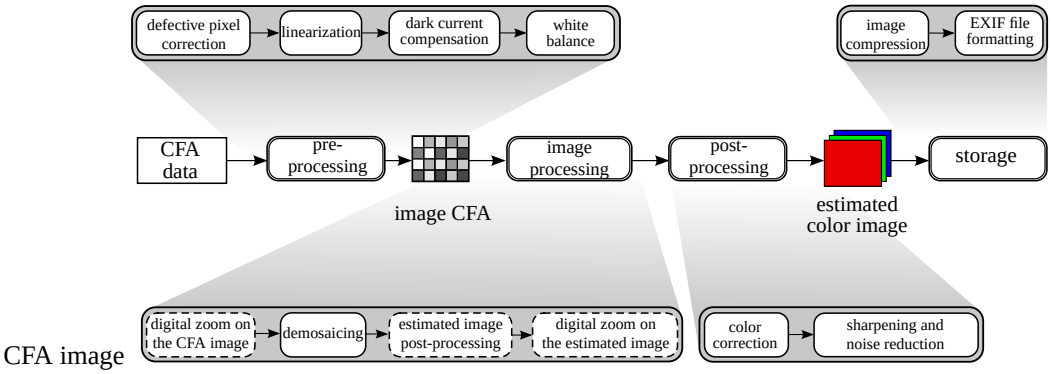
As shown in figure 3b, many other processing tasks are classically achieved within a mono-CCD color camera (Lukac and Plataniotis, 2007). They consist for instance in raw sensor data correction or, after demosaicing, in color improvement, image sharpening and noise reduction, so as to provide a "visually pleasing" color image to the user. These processing tasks are essential to the quality of the provided image and, as a matter of fact, discriminate the various models of digital cameras, since manufacturers and models of sensors are not so numerous. The related underlying algorithms have common features or basis, and parameter tuning is often a key step leading to more or fewer residual errors. Together with noise characteristics of the imaging sensor, such *artifacts* may incidentally be used to typify each camera model (Bayrama et al., 2008).

2.1.2. Color Filter Arrays

Several configurations may be considered for the CFA, and figure 4 shows some examples found in the literature. A few mono-CCD cameras use a CFA based on complementary color components (Cyan, Magenta and Yellow), with a 2×2 pattern which also sometimes includes a filter sensitive to the green light. But the very large majority of cameras are equipped with filter arrays based on R, G and B primary color components. Regardless of their arrangement and design, these arrays often include twice as many filters sensitive to the green primary as filters sensitive to blue or red light. This stems from Bayer's observation that the human eye has a greater resolving power for green light. Moreover, the photopic *luminous efficiency function* of the human retina – also known as the *luminosity function* – is similar to the CIE 1931 green matching function $G_c(\lambda)$, with a maximum reached in the same spectral domain. Bayer

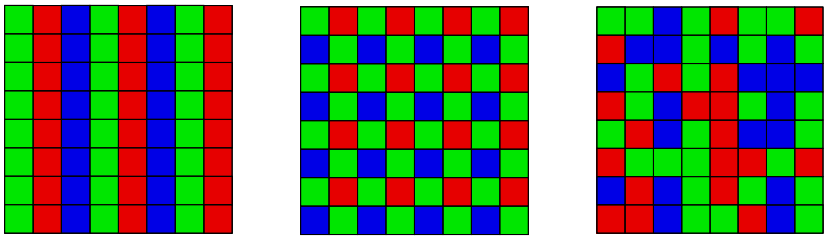


(a) Mono-CCD technology outline, using the Bayer Color Filter Array (CFA).



(b) Image acquisition within a mono-CCD color camera (detailed schema). Dotted steps are optional.

FIG. 3: Internal structure of a mono-CCD color camera.



(a) Vertical stripes (b) Bayer (c) Pseudo-random

(d) Complementary colors (e) "Panchromatic", or CFA2.0 (Kodak) (f) "Burtoni" CFA

FIG. 4: Configuration examples for the mosaic of color filters. Each square depicts a pixel in the CFA image, and its color is that of the monochromatic filter covering the associated photosite.

therefore both makes the assumption that green photosensors capture luminance, whereas red and blue ones capture chrominance, and suggests to fill the CFA with more luminance-sensitive (green) elements than chrominance-sensitive (red and blue) elements (see figure 4b).

The CFA using alternating vertical stripes (see figure 4a) of the *RGB* primaries has been released first, since it is well suited to the interlaced television video signal. Nevertheless, considering the Nyquist limits for the green component plane, Parulski (1985) shows that the Bayer CFA has larger bandwidth than the latter for horizontal spatial frequencies. The pseudo-random filter array (see figure 4c) has been inspired by the human eye physiology, in an attempt to reproduce the spatial repartition of the three cone cell types on the retina surface (Lukac and Plataniotis, 2005a). Its irregularity achieves a compromise between the sensitivity to spatial variations of luminance in the observed scene (*visual acuity*) and the ability to perceive thin objects with different colors (Roorda et al., 2001). Indeed, optimal visual acuity would require photosensors with identical spectral sensitivities which are constant over the spectrum, whereas the perception of thin color objects is better ensured with sufficient local density of different types of cones. Despite pseudo-random color filter arrays show interesting properties (Alleysson et al., 2008), their design and exploitation have not much been investigated so far ; for some discussions, see e.g. Condat (2009) or Savard (2007) about CFA design and Zapryanov and Nikolova (2009) about demosaicing of Bayer CFA "pseudo-random" variations. Among other studies drawing their inspiration from natural physiology for CFA design, Kröger's work (2004) yields a new mo-

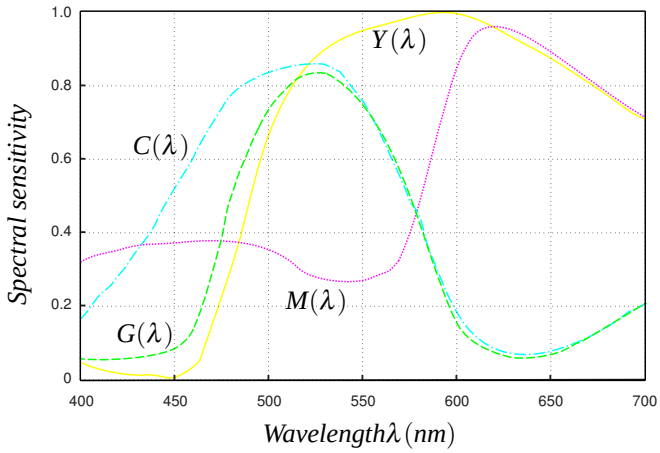


FIG. 5: Relative spectral sensitivity of the JAI CV-S3300P camera sensor (Jai Corporation, 2000).

saic which mimics the retina of a cichlid fish, *Astatotilapia burtoni* (Günther, 1894). It is shown that this particular arrangement (see figure 4f), which includes many spatial frequencies and different geometries for color components, generates weak aliasing artifacts. This complex mosaic configuration efficiently enhances the simulated image quality (Medjeldi et al., 2009), but the effective implementation of such a sensor, and the demosaicing step of the corresponding CFA images, are open and challenging problems.

Color filter arrays based on complementary primary colors have also been designed and used, with two main advantages. First, they own higher spectral sensitivity and wider bandwidth than RGB filters, which is of particular interest in noisy environments and/or when the frame rate imposes low integration period (Hirakawa, 2008). Figure 5 shows the spectral sensitivity of the JAI CV-S3300P camera sensor, equipped with the CFA of figure 4d. A few years ago, some professional still cameras used complementary color filter arrays to ensure high ISO sensitivity, as the Kodak DCS-620x model equipped with a CMY filter (Noble, 2000). As a second advantage, these CFAs make the generation of television luminance/chroma video signal almost immediate, and are sometimes embedded in PAL or NTSC color video cameras (Sony Corporation, 2000). Their usage is however largely restricted to television, since the strong mutual overlapping of C,M,Y spectral sensitivity functions makes the conversion into R,G,B primaries unsatisfactory.

New types of CFA have recently been released, and used in camera models released by two major manufacturers. Since 1999, Fuji develops a new so-called *Super CCD* sensor, based on photosites in a 45-degree oriented honeycomb lattice (see figure 6). The HR version of 2003 (see figure 6a) allows to optimize the occupancy on the CCD surface, hence to potentially capture more light. “Square” pixels are obtained from octagonal photosites by combining the four neighbors in part, so that new pixels are created and the resolution is doubled. An alternative version of this sensor

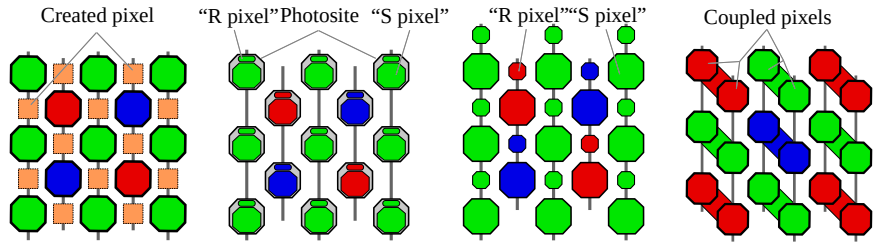


FIG. 6: Super CCD technology. For clarity sake, photosites are represented further apart from each other than at their actual location.

(SR, see figure 6b) has expanded dynamic range, by incorporating both high-sensitivity large photodiodes (“S-pixels”) used to capture normal and dark details, and smaller “R-pixels” sensitive to bright details. The EXR version (see figure 6d) takes advantage of same idea, but extra efforts have been conducted on noise reduction thanks to pixel binning, resulting in a new CFA arrangement and its exploitation by pixel coupling. As a proprietary technology, little technical detail is available on how Super CCD sensors turn the image into an horizontal/vertical grid without interpolating, or on how demosaicing associated with such sensors is achieved. A few hints may however be found in a patent using a similar imaging device (Kuno and Sugiura, 2006).

In 2007, Kodak develops new filter arrays (Hamilton and Compton, 2007) as another alternative to the widely used Bayer CFA. The basic principle of this so-called CFA2.0 family of color filters is to incorporate transparent filter elements (represented as white squares on figure 4e), those filters being hence also known as *RGBW* or “panchromatic” ones. This property makes the underlying photosites sensitive to all wavelengths of the visible light. As a whole, the sensors associated with CFA2.0 are therefore more sensitive to low-energy stimuli than those using Bayer CFA. Such increase of global sensitivity leads to better luminance estimation, but at the expense of chromatic information estimation. Figure 7 shows the processing steps required to estimate a full color image from the data provided by a CFA2.0-based sensor.

By modifying the CFA arrangement, manufacturers primarily aim at increasing the spectral sensitivity of the sensor. Lukac and Plataniotis (2005a) tackled the CFA design issue by studying the influence of the CFA configuration on demosaicing results. They considered ten different RGB color filter arrays, three of them being shown on figures 4a to 4c. A CFA image is first simulated by sampling one out of the three color components at each pixel in an original color image, according to the considered CFA pattern. A universal demosaicing framework is then applied to obtain a full-color image. The quality of the demosaiced image is finally evaluated by comparing it to the original image thanks to several objective error criteria. The authors conclude that the CFA design is critical to demosaicing quality results, but cannot advise any CFA that would yield best results in all cases. Indeed, the relative performance of filters is highly dependent on the tested image.

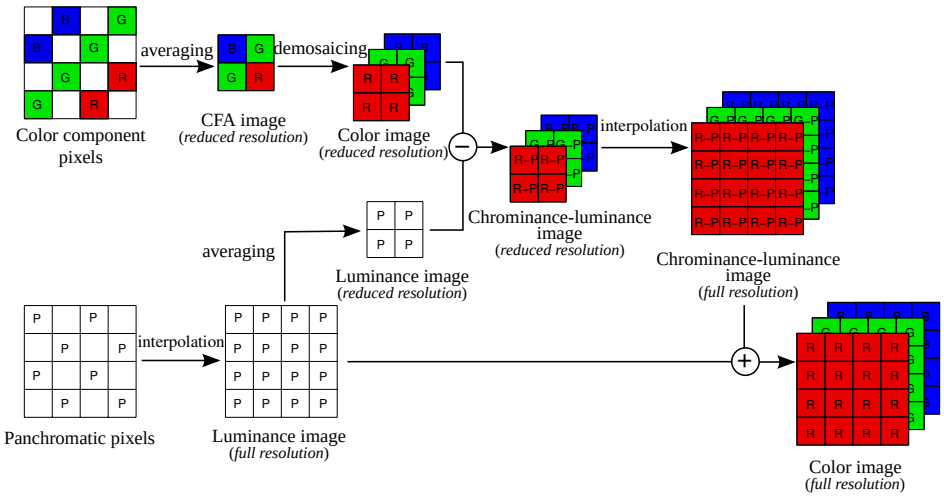


FIG. 7: Processing steps of the raw image provided by a CFA2.0-based sensor. "Panchromatic pixels" are those associated with photosites covered with transparent filters.

All in all, the Bayer CFA achieves a good compromise between horizontal and vertical resolutions, luminance and chrominance sensitivities, and therefore remains the favorite CFA in industrial applications. As this CFA is the most commonly used and has inspired some more recent ones, it will be considered first and foremost in the following text. Demosaicing methods presented hereafter are notably based on the Bayer CFA.

2.1.3. Demosaicing Formalization

Estimated colors have less fidelity to color stimuli from the observed scene than those provided by a three-CCD camera. Improving the quality of color images acquired by mono-CCD cameras is still a highly relevant topic, investigated by researchers and engineers (Lukac, 2008). In this paper, we focus on the demosaicing step and examine its influence on the estimated image quality.

In order to set a formalism for the demosaicing process, let us compare the acquisition process of a color image in a three-CDD camera and in a mono-CCD camera. Figure 8a outlines a three-CCD camera architecture, in which the color image of a scene is formed by combining the data from three sensors. The resulting color image $\mathbf{I}$ is composed of three color component planes I^k, $k \in \{R,G,B\}$. In each plane I^k, a given pixel P is characterized by the level of the color component k. A three-component vector defined as $\mathbf{I}_{x,y} \triangleq (R_{x,y},G_{x,y},B_{x,y})$ is therefore associated with each pixel – located at spatial coordinates (x,y) in image $\mathbf{I}$. In a color mono-CCD camera, the color image generation is quite different, as shown in figure 8b : the single sensor delivers a raw image, hereafter called *CFA image* and denoted I^{CFA}. If the Bayer CFA is considered, to each pixel with coordinates (x,y) in image I^{CFA} is associated a single color

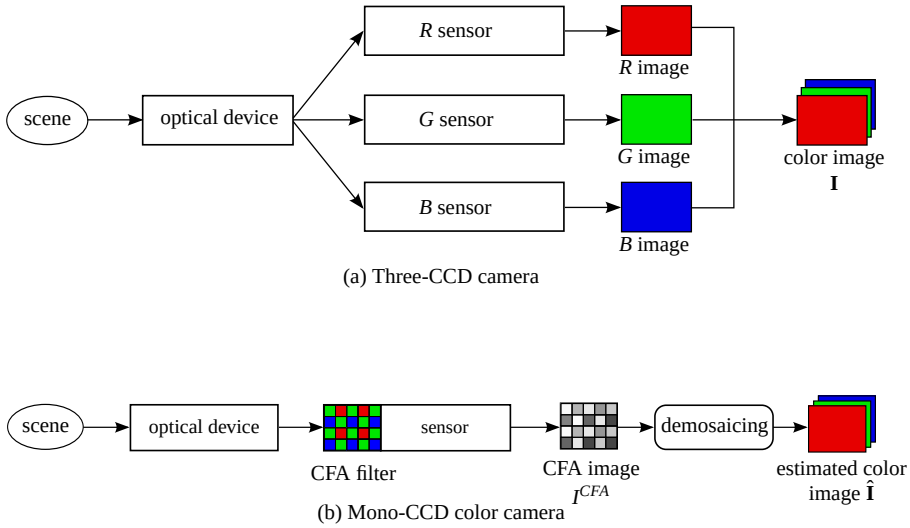


FIG. 8: Color image acquisition outline, according to the camera type.

component R, G or B (see figure 9) :

$$I_{x,y}^{CFA} = \begin{cases} R_{x,y} & \text{if } x \text{ is odd and } y \text{ is even,} \quad (1a) \\ B_{x,y} & \text{if } x \text{ is even and } y \text{ is odd,} \quad (1b) \\ G_{x,y} & \text{otherwise.} \quad (1c) \end{cases}$$

The color component levels range from 0 to 255 when they are quantized with 8 bits.

The demosaicing scheme $\mathscr{F}$, most often implemented as an interpolation procedure, consists in estimating a color image $\hat{\mathbf{I}}$ from I^{CFA}. At each pixel of the estimated image, the color component available in I^{CFA} at the same pixel location is picked up, whereas the other two components are estimated :

$$I_{x,y}^{CFA} \xrightarrow{\mathscr{F}} \hat{\mathbf{I}}_{x,y} = \begin{cases} (R_{x,y}, \hat{G}_{x,y}, \hat{B}_{x,y}) & \text{if } x \text{ is odd and } y \text{ is even,} \quad (2a) \\ (\hat{R}_{x,y}, \hat{G}_{x,y}, B_{x,y}) & \text{if } x \text{ is even and } y \text{ is odd,} \quad (2b) \\ (\hat{R}_{x,y}, G_{x,y}, \hat{B}_{x,y}) & \text{otherwise.} \quad (2c) \end{cases}$$

Each triplet in equations (2) stands for a color, whose color component available at pixel $P(x,y)$ in I^{CFA} is denoted $R_{x,y}$, $G_{x,y}$ or $B_{x,y}$, and whose other two components among $\hat{R}_{x,y}$, $\hat{G}_{x,y}$ and $\hat{B}_{x,y}$ are estimated for $\hat{\mathbf{I}}_{x,y}$.

Before we get to the heart of the matter, let us still precise a few notations that will be most useful later in this section. In the CFA image (see figure 9), four different

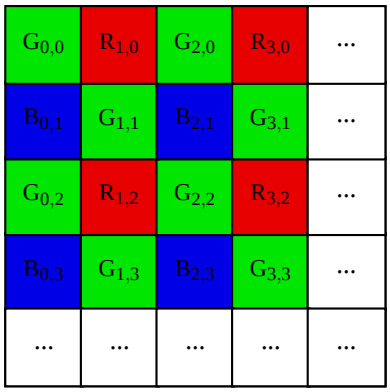
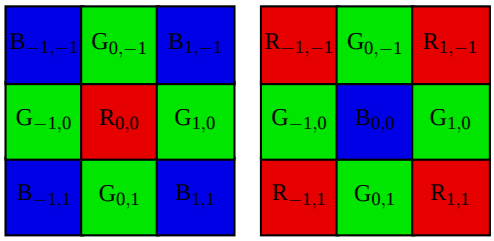
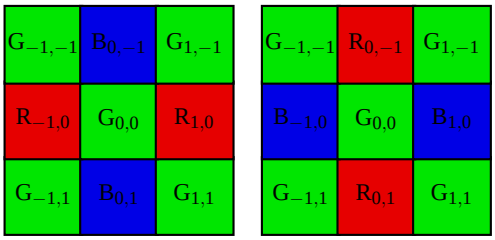


FIG. 9: CFA image from the Bayer filter. Each pixel is artificially colorized with the corresponding filter main spectral sensitivity, and the presented arrangement is the most frequently encountered in the literature (i.e. *G* and *R* levels available for the first two row pixels).



(a) $\{GRG\}$ (b) $\{GBG\}$



(c) $\{RGR\}$ (d) $\{BGB\}$

FIG. 10: 3×3 neighborhood structures of pixels in the CFA image.

structures are encountered for the 3×3 spatial neighborhood, as shown on figure 10. For each of these structures, the pixel under consideration for demosaicing is the central one, at which the two missing color components should be estimated thanks to the available components and their levels at the neighboring pixels. Let us denote the aforementioned structures by the color components available on the middle row, namely $\{GRG\}$, $\{GBG\}$, $\{RGR\}$ and $\{BGB\}$. Notice that $\{GRG\}$ and $\{GBG\}$ are structurally similar, apart from the slight difference that components R and B are exchanged. Therefore, they can be analyzed in the same way, as can $\{RGR\}$ and $\{BGB\}$ structures. A generic notation is hence used in the following : the center pixel is considered having $(0,0)$ spatial coordinates, and its neighbors are referred to using their relative coordinates $(\delta x,\delta y)$. Whenever this notation bears no ambiguity, $(0,0)$ coordinates are omitted. Moreover, we also sometimes use a letter (e.g. P) to generically refer to a pixel, its color components being then denoted as $R(P)$, $G(P)$ and $B(P)$. The notation $P(\delta x,\delta y)$ allows to refer to a pixel thanks to its relative coordinates, its colors components being then denoted $R_{\delta x,\delta y}$, $G_{\delta x,\delta y}$ and $B_{\delta x,\delta y}$, as in figure 10.

2.1.4. Demosaicing Evaluation Outline

Demosaicing objective is to generate an estimated color image $\hat{\mathbf{I}}$ as close as possible to the original image $\mathbf{I}$. Even this image is unavailable effectively, $\mathbf{I}$ is generally used as a reference to evaluate the demosaicing quality. Then, one either strive to obtain as a low value as possible for an error criterion, or as a high value as possible for a quality criterion comparing the estimated image and the original one. A classical evaluation procedure for the demosaicing result quality consists in (see figure 11) :

1. simulating a CFA image provided by a mono-CCD camera from a color original image provided by a three-CCD camera. This is achieved by sampling a single color component R, G or B at each pixel, according to the considered CFA arrangement (Bayer CFA of figure 9, in our case) ;
2. demosaicing this CFA image to obtain an estimated color image ;
3. comparing the original and estimated color images, so as to highlight artifacts affecting the latter.

There is no general agreement on the demosaicing quality definition, which is highly dependent upon the estimated color image exploitation – as will be detailed in the next sections. In a first time, we will rely on visual examination, or else on the most used quantitative criterion (signal-to-noise ratio) for a quality result evaluation, which both require a reference image. As in most works related to demosaicing, we will here use the Kodak image database (Kodak, 1991) as a benchmark for performance comparison of the various methods, as well as for illustration purposes. More precisely, to avoid overloaded results, a representative subset of twelve of these images has been picked up as the most used set in literature. These natural images contain rich colors and textural regions, and are fully reproduced in figure 37 so that they can be referred to in the text.

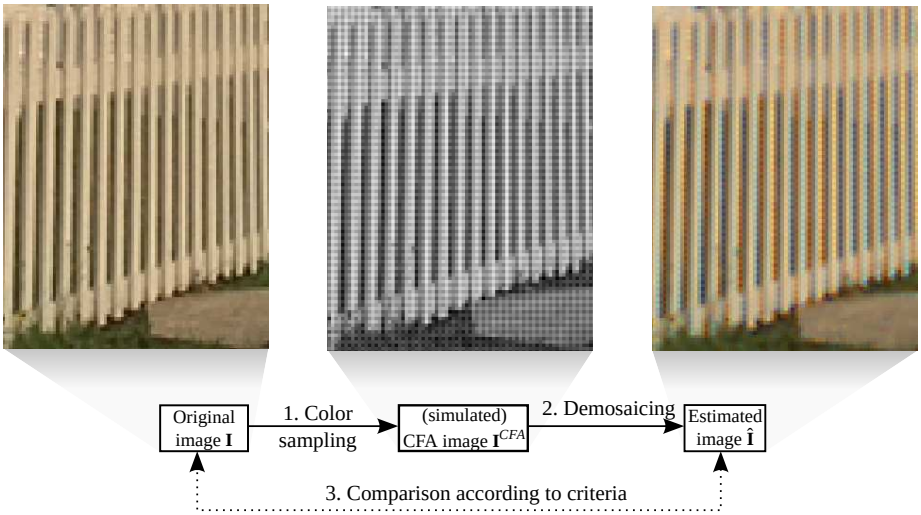


FIG. 11: Classical evaluation procedure for the demosaicing result quality (example of bilinear interpolation on an extract from the Kodak benchmark image "Lighthouse").

2.2. Basic Schemes and Demosaicing Rules

2.2.1. Bilinear Interpolation

The first solutions for demosaicing were proposed in the early eighties. They process each component plane separately and find the missing levels by applying linear interpolation on the available ones, in both main directions of the image plane. Such a bilinear interpolation is traditionally used to resize gray-level images (Gribbon and Bailey, 2004). Considering the $\{GRG\}$ structure, the missing blue and green values at the center pixel are respectively estimated by bilinear interpolation thanks to the following equations :

$$\hat{B} = \frac{1}{4}\left(B_{-1,-1}+B_{1,-1}+B_{-1,1}+B_{1,1}\right), \quad (3)$$

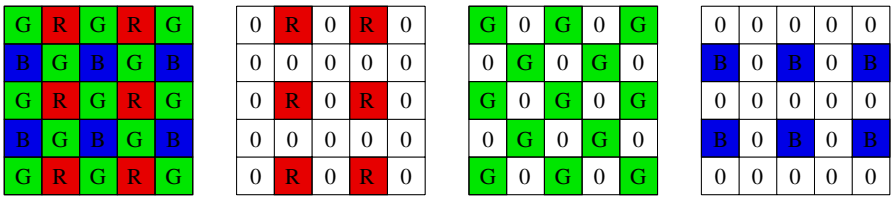
$$\hat{G} = \frac{1}{4}\left(G_{0,-1}+G_{-1,0}+G_{1,0}+G_{0,1}\right). \quad (4)$$

As for the $\{RGR\}$ structure, the missing red and blue component levels are estimated as follows :

$$\hat{R} = \frac{1}{2}\left(R_{-1,0}+R_{1,0}\right), \quad (5)$$

$$\hat{B} = \frac{1}{2}\left(B_{0,-1}+B_{0,1}\right). \quad (6)$$

Alleysson et al. (2008) notice that such interpolation is achievable by convolution. For that purpose, consider the three planes formed of the sole levels of component k, $k \in \{R,G,B\}$, available in the CFA image, other component levels being set to zero. Let



(a) I^{CFA} (b) $\varphi^R\left(I^{CFA}\right)$ (c) $\varphi^G\left(I^{CFA}\right)$ (d) $\varphi^B\left(I^{CFA}\right)$

FIG. 12: Definition of planes $\varphi^k\left(I^{CFA}\right)$ by sampling the CFA image according to each color component k, $k \in \{R,G,B\}$. The CFA image and planes $\varphi^k\left(I^{CFA}\right)$ are here colorized for illustration sake.

us denote $\varphi^k(I)$ the function sampling a gray-level image I according to the locations of the available color component k in the CFA :

$$\varphi^k(I)(x,y) = \begin{cases} I(x,y) & \text{if component } k \text{ is available at pixel } P(x,y) \text{ in } I^{CFA}, \\ 0 & \text{otherwise.} \end{cases} \quad (7)$$

Figure 12 illustrates the special cases of planes $\varphi^k(I^{CFA})$ obtained by applying functions φ^k to I^{CFA}.

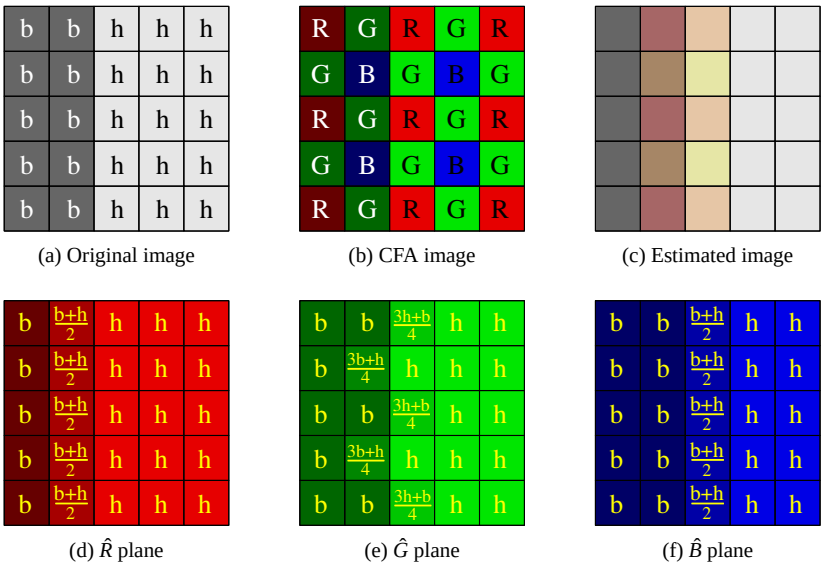
Let us also consider the convolution filters defined by the following kernels :

$$H^R = H^B = \frac{1}{4}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \quad (8) \qquad \text{and} \qquad H^G = \frac{1}{4}\begin{bmatrix} 0 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 0 \end{bmatrix} . \quad (9)$$

In order to determine the color image $\hat{\mathbf{I}}$, each color component plane $\hat{I}^k$ can now be estimated by applying the convolution filter of kernel H^k on the plane $\varphi^k\left(I^{CFA}\right)$, respectively :

$$\hat{I}^k = H^k * \varphi^k(I^{CFA}) \text{ , } k \in \{R,G,B\} \text{ .} \quad (10)$$

Bilinear interpolation is easy to be implemented and not processing time consuming, but generates severe visible artifacts, as also shown in figure 11. The above scheme provides satisfying results in image areas with homogeneous colors, but many false colors in areas with spatial high frequencies – as for the fence bars in this extract. Following Chang and Tan (2006), a deep study of the causes of theses artifacts can be achieved by simulating their generation on a synthetic image (see figure 13a). In this original image, two homogeneous areas are separated by a vertical transition, which recreates the boundary between two real objects with different gray levels. At each pixel, the levels of all three color components are then equal. Levels of pixels depicting the darker left object (labeled as b) are lower than those of pixels depicting the lighter right object (labeled as h). Figure 13b shows the CFA image I^{CFA} yielded by sampling



(a) Original image (b) CFA image (c) Estimated image

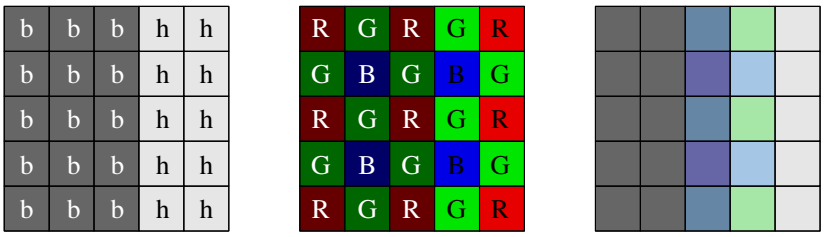
(d) $\hat{R}$ plane (e) $\hat{G}$ plane (f) $\hat{B}$ plane

FIG. 13: Demosaicing by bilinear interpolation of an gray-level image with a vertical transition. The CFA image and $\hat{R}$, $\hat{G}$ and $\hat{B}$ planes are here colorized for illustration sake.

a single color component per pixel according to the Bayer CFA. The result of bilinear interpolation demosaicing applied to this image is given on figure 13c. Figures 13d to 13f give details on the three estimated color planes $\hat{R}$, $\hat{G}$ and $\hat{B}$. On $\hat{R}$ and $\hat{B}$ planes, this demosaicing algorithm generates a column of intermediate-level pixels, whose value is the average of the two object levels. On the green plane, it produces a jagged pattern on both edge sides, formed of pixels alternating between two intermediate levels – a low one $(3b+h)/4$ and a high one $(3h+b)/4$. As a whole, the edge area is formed of a square 2×2 pattern of four different colors repeated alongside the transition (see the estimated image in figure 13c). This demosaicing procedure has hence generated two types of artifacts : erroneously estimated colors (hereafter referred to as "false colors"), and an artificial jagged pattern (so-called "zipper effect"), which are both studied in section 4.2. According to the horizontal transition location relative to the CFA mosaic, the generated pattern may be either orange-colored as in figure 13c or with bluish colors as in figure 14c. These two dominant-color patterns may be actually observed in the estimated image of figure 11.

2.2.2. Main Demosaicing Rules

Let us examine the component-wise profiles of the middle pixel row in the original image 13a and its corresponding estimated image 13c. Dissimilarities between these profiles on R, G and B planes are underlined on figure 15 : the transition occurs at identical horizontal locations on the three original image planes, but this is no more the case for the estimated image. Such inconsistency among the demosaicing results for different components generates false colors in the estimated image formed from



(a) Refrence image (b) CFA image (c) Estimated image

FIG. 14: Variant version of image 13a, demosaiced by bilinear interpolation as well.

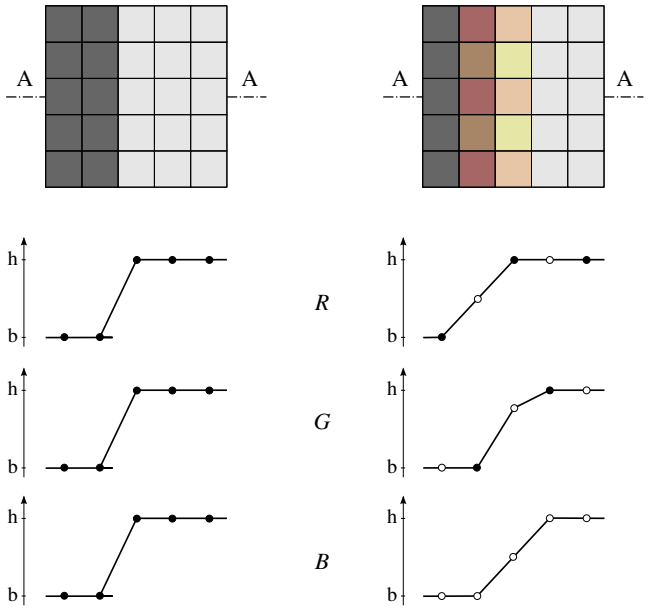
their combination. It can also be noticed that the transition corresponds, in each color plane of the original image, to a local change of homogeneity along the horizontal direction. Bilinear interpolation averages the levels of pixels located on both sides of the transition, which makes the latter less sharp.

In accordance with the previous observations, we can state that two main rules have to be enforced so as to improve demosaicing results : spatial correlation and spectral correlation.

– Spectral correlation.
The transition profiles plotted in figure 15 are identical for the original image component planes, which conveys strict correlation between components. For a natural image, Gunturk et al. (2002) show that the three color components are also strongly correlated. The authors apply a bidimensional filter built on a low-pass filter $h_0 = [1\ 2\ 1]/4$ and a high-pass one $h_1 = [1\ -2\ 1]/4$, so as to split each color component plane into four subbands resulting from row and column filtering : (LL) both rows and columns are low-pass filtered ; (LH) rows are low-pass and columns high-pass filtered ; (HL) rows are high-pass and columns low-pass filtered ; (HH) both rows and columns are high-pass filtered. For each color component, four subband planes are obtained in this way, respectively representing data in rather homogeneous areas (low-frequency information), horizontal detail (high-frequency information in the horizontal direction), vertical detail (high-frequency information in the vertical direction) and diagonal detail (high-frequency information in both main directions). The authors then compute a correlation coefficient $r^{R,G}$ between red and green components over each subband according to the following formula :

$$r^{R,G} = \frac{\sum_{x=0}^{X-1}\sum_{y=0}^{Y-1}\left(R_{x,y}-\mu^R\right)\left(G_{x,y}-\mu^G\right)}{\sqrt{\sum_{x=0}^{X-1}\sum_{y=0}^{Y-1}\left(R_{x,y}-\mu^R\right)^2}\sqrt{\sum_{x=0}^{X-1}\sum_{y=0}^{Y-1}\left(G_{x,y}-\mu^G\right)^2}}, \tag{11}$$

in which $R_{x,y}$ (respectively $G_{x,y}$) is the level at (x,y) pixel in the red (respectively green) component plane within the same subband, μ^R and μ^G being the average of $R_{x,y}$ and $G_{x,y}$ levels over the same subband planes. The correlation coefficient between the blue and green components is similarly computed. Test results on



(a) Original image (b) Estimated image

FIG. 15: Component-wise profiles of middle pixel row levels A-A in the original and estimated images. Black dots stand for available levels, and white dots for estimated levels.

twenty natural images show that those coefficients are always greater than 0.9 in subbands carrying spatial high frequencies at least in one direction (i.e. LH, HL and HH). As for the subband carrying low frequencies (LL), coefficients are lower but always greater than 0.8. This reveals a very strong correlation between levels of different color components in a natural image, especially in areas with high spatial frequencies. Lian et al. (2006) confirm, using a wavelet coefficient analysis, that high-frequency information is not only strongly correlated between the three component planes, but almost identical. Such *spectral correlation* between components should be taken into account to retrieve the missing components at a given pixel.

– Spatial correlation.
  A color image can be viewed as a set of adjacent homogeneous regions whose pixels have similar levels for each color component. In order to estimate the missing levels at each considered pixel, one therefore should exploit the levels of neighboring pixels. However, this task is difficult at pixels near the border between two distinct regions due to high local variation of color components. As far as demosaicing is concerned, this *spatial correlation* property avoids to interpolate missing components at a given pixel thanks to neighbor levels which do not belong to the same homogeneous region.

These two principles are generally taken into account sequentially by the demosaicing procedure. In the first step, demosaicing often consists in estimating the green component using spatial correlation. According to Bayer's assumption, the green component has denser available data within the CFA image, and represents the luminance of the image to be estimated. Estimation of red and blue components (assimilated to chrominance) is only achieved in a second step, thanks to the previously interpolated luminance and using the spectral correlation property. Such a way of using both correlations is used by a large number of methods in the literature. Also notice that, although red and blue component interpolation is achieved after the green plane has been fully populated, spectral correlation is also often used in the first demosaicing step to improve the green plane estimation quality.

2.2.3. Spectral Correlation Rules

In order to take into account the strong spectral correlation between color components at each pixel, two main hypotheses are proposed in the literature. The first one assumes a color *ratio* constancy and the second one is based on color *difference* constancy. Let us examine the underlying principles of each of these assumptions before comparing both.

Interpolation based on color hue constancy, suggested by Cok (1987), is historically the first one based on spectral correlation. According to Cok, *hue* is understood as the ratio between chrominance and luminance, i.e. R/G or B/G. His method proceeds in two steps. In the first step, missing green values are estimated by bilinear interpolation. Red (and blue) levels are then estimated by weighting the green level at the given pixel with the hue average of neighboring pixels. For instance, interpolation of the blue level at the center pixel of $\{GRG\}$ CFA structure (see figure 10a) uses the four diagonal

neighbors where this blue component is available :

$$\hat{B} = \hat{G} \cdot \frac{1}{4} \left[\frac{B_{-1,-1}}{\hat{G}_{-1,-1}} + \frac{B_{1,-1}}{\hat{G}_{1,-1}} + \frac{B_{-1,1}}{\hat{G}_{-1,1}} + \frac{B_{1,1}}{\hat{G}_{1,1}} \right]. \tag{12}$$

This bilinear interpolation between color component ratios is based on the local constancy of this ratio within an homogeneous region. Kimmel (1999) justifies the color ratio constancy assumption thanks to a simplified approach that models any color image as a *Lambertian* object surface observation. According to the Lambertian model, such a surface reflects the incident light to all directions with equal energy. The intensity $I(P)$ received by the photosensor element associated to each pixel P is therefore independent of the camera position, and can be represented as :

$$I(P) = \rho \left\langle \vec{N}(P), \vec{l} \right\rangle, \tag{13}$$

where ρ is the albedo (or reflection coefficient), $\vec{N}(P)$ is the normal vector to the surface element which is projected on pixel P, and $\vec{l}$ is the incident light vector. As the albedo ρ characterizes the object material, this quantity is different for each color component ($\rho^R \neq \rho^G \neq \rho^B$), and the three color components may be written as :

$$I^R(P) = \rho^R \left\langle \vec{N}(P), \vec{l} \right\rangle, \tag{14}$$

$$I^G(P) = \rho^G \left\langle \vec{N}(P), \vec{l} \right\rangle, \tag{15}$$

$$I^B(P) = \rho^B \left\langle \vec{N}(P), \vec{l} \right\rangle. \tag{16}$$

Assuming that any object is composed of one single material, coefficients ρ^R, ρ^G and ρ^B are then constant at all pixels representing an object. So, the ratio between two color components is also constant :

$$K^{k,k'} = \frac{I^k(P)}{I^{k'}(P)} = \frac{\rho^k \left\langle \vec{N}(P), \vec{l} \right\rangle}{\rho^{k'} \left\langle \vec{N}(P), \vec{l} \right\rangle} = \frac{\rho^k}{\rho^{k'}} = constant, \tag{17}$$

where $(k,k') \in \{R,G,B\}^2$. Although this assumption is simplistic, it is locally valid and can be used within the neighborhood of the considered pixel.

Another simplified and widely used model of correlation between components relies on the color *difference* constancy assumption. At a given pixel, this can be written as :

$$D^{k,k'} = I^k(P) - I^{k'}(P) = \rho^k \left\langle \vec{N}(P), \vec{l} \right\rangle - \rho^{k'} \left\langle \vec{N}(P), \vec{l} \right\rangle = constant, \tag{18}$$

where $(k,k') \in \{R,G,B\}^2$. As the incident light direction and amplitude are assumed to be locally constant, the color component difference is also constant within the considered pixel neighborhood.

As a consequence, the chrominance interpolation step in Cok's method may be rewritten by using component *difference* averages, for instance :

$$\hat{B} = \hat{G} + \frac{1}{4}\left[(B_{-1,-1} - \hat{G}_{-1,-1}) + (B_{1,-1} - \hat{G}_{1,-1}) + (B_{-1,1} - \hat{G}_{-1,1}) + (B_{1,1} - \hat{G}_{1,1})\right], \tag{19}$$

instead of equation (12). The validity of this approach is also justified by Lian et al. (2007) on the ground of spatial high frequency similarity between color components.

The color difference constancy assumption is globally consistent with the ratio rule used in formula (12). By considering the logarithmic non-linear transformation, the difference $D_2^{k,k'}, (k,k') \in \{R,G,B\}^2$, can be expressed as :

$$D_2^{k,k'} = \log_{10}\left(\frac{I^k(P)}{I^{k'}(P)}\right) = \log_{10}\left(I^k(P)\right) - \log_{10}\left(I^{k'}(P)\right). \tag{20}$$

Furthermore, we propose to compare those two assumptions expressed by equations (17) and (18). In order to take into account spectral correlation for demosaicing, it turns out that the difference of color components presents some benefits in comparison to their ratio. The latter is indeed error-prone when its denominator takes low values. This happens for instance when saturated red and/or blue components lead to comparatively low values of green, making the ratios in equation (12) very sensitive to red and/or blue blue small variations. Figure 16a is a natural image example which is highly saturated in red. Figures 16c and 16d show the images where each pixel value is, respectively, the component ratio R/G and difference $R-G$ (pixel levels being normalized by linear dynamic range stretching). It can be noticed that these two images actually carry out less high-frequency information than the green component plane shown on figure 16b.

A Sobel filter is then applied to these two images, so as to highlight the high-frequency information location. The Sobel filter output module is shown on figures 16e and 16f. In the right-hand parrot plumage area where red is saturated, the component ratio plane contains more high-frequency information than the component difference plane, which makes it more artifact-prone when demosaiced by interpolation. Moreover, high color ratio values may yield to estimated component levels beyond the data bounds, which is undesirable for the demosaicing result quality.

To overcome these drawbacks, a linear translation model applied on all three color components is suggested by Lukac and Plataniotis (2004a, 2004b). Instead of equation (17), the authors reformulate the color ratio rule by adding a predefined constant value β to each component. The new constancy assumption, which is consistent with equation (17) in homogeneous areas, now relies on the ratio :

$$K_2^{k,k'} = \frac{I^k + \beta}{I^{k'} + \beta}, \tag{21}$$

where $(k,k') \in \{R,G,B\}^2$, and where $\beta \in \mathbb{N}$ is a ratio normalization parameter. Under this new assumption on the normalized ratio, the blue level interpolation formulated in

(a) Original image

(b) G plane

(c) R/G ratio plane

(d) $R-G$ difference plane

(e) Sobel filter output on the R/G plane

(f) Sobel filter output on the $R-G$ plane

FIG. 16: Component ratio and difference planes on a same image (“Parrots” from the Kodak database).

equation (12) under the ratio rule now becomes[1] :

$$\hat{B} = -\beta + \left(\hat{G} + \beta\right) \cdot \frac{1}{4} \cdot \left[\frac{B_{-1,-1} + \beta}{\hat{G}_{-1,-1} + \beta} + \frac{B_{1,-1} + \beta}{\hat{G}_{1,-1} + \beta} + \frac{B_{-1,1} + \beta}{\hat{G}_{-1,1} + \beta} + \frac{B_{1,1} + \beta}{\hat{G}_{1,1} + \beta} \right]. \quad (22)$$

In order to avoid too different values for the numerator and denominator, Lukac and Plataniotis advise to set $\beta = 256$, so that the normalized ratios R/G and B/G range from 0.5 to 2. They claim that this assumption improves the interpolation quality in areas of transitions between objects and of thin details.

In our investigation of the two main assumptions used for demosaicing, we finally compare the estimated image quality in both cases. The procedure depicted on figure 11 is applied on twelve natural images selected from Kodak database : the demosaicing schemes presented above, respectively using component ratio and difference, are applied to the simulated CFA image. To evaluate the estimated color image quality in comparison with the original image, we then compute an objective criterion, namely the peak signal-to-noise ratio (*PSNR*) derived from the mean square error (*MSE*) between the two images. On the red plane for instance, these quantities are defined as :

$$MSE^R = \frac{1}{XY} \sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} \left(I^R_{x,y} - \hat{I}^R_{x,y}\right)^2, \quad (23)$$

$$PSNR^R = 10 \cdot \log_{10} \left(\frac{255^2}{MSE^R} \right). \quad (24)$$

As the green component is bilinearly interpolated without using spectral correlation, only red and blue estimated levels vary according to the considered assumption. The *PSNR* is hence computed on these two planes. Results displayed in table 1 show that using the color difference assumption yields better results than using the simple ratio rule K, which is particularly noticeable for image “Parrots” of figure 16a. The normalized ratio K_2, which is less prone to large variations than K in areas with spatial high frequencies, leads to higher values for $PSNR^R$ and $PSNR^B$. However, the color difference assumption generally outperforms ratio-based rules according to the *PSNR* criterion, and is most often used to exploit spectral correlation in demosaicing schemes.

[1] The authors use, in this interpolation formula, extra weighting factors depending on the local pattern and dropped here for conciseness.

Image	$PSNR^R$			$PSNR^B$		
	D	K	K_2	D	K	K_2
1 (“Parrots”)	**38.922**	36.850	38.673	38.931	38.678	**38.936**
2 (“Sailboats”)	**31.321**	31.152	31.311	**31.154**	30.959	31.129
3 (“Windows”)	**37.453**	36.598	37.348	**37.093**	36.333	36.676
4 (“Houses”)	27.118	26.985	**27.146**	27.007	26.889	**27.008**
5 (“Race”)	**36.085**	35.838	36.073	**35.999**	35.819	35.836
6 (“Pier”)	**32.597**	31.911	32.563	**32.570**	32.178	32.217
7 (“Island”)	**34.481**	34.376	34.470	**34.402**	34.208	34.399
8 (“Lighthouse”)	**31.740**	31.415	31.696	**31.569**	31.093	31.289
9 (“Plane”)	**35.382**	35.058	35.347	**34.750**	34.324	34.411
10 (“Cape”)	**32.137**	31.863	32.118	**31.842**	31.532	31.693
11 (“Barn”)	**34.182**	33.669	34.143	**33.474**	33.193	33.363
12 (“Chalet”)	**30.581**	30.413	30.565	**29.517**	29.263	29.364
Average	**33.500**	33.011	33.454	**33.192**	32.872	33.027

TAB. 1: Peak signal-to-noise ratios (in decibels) for red ($PSNR^R$) and blue ($PSNR^B$) planes of twelve Kodak images (Eastman Kodak and various photographers, 1991), demosaiced under the color difference D (see equation (18) and interpolation formula (19)), under the color ratio K (see equation (17) and interpolation formula (12)) and under the normalized ratio K_2 ($\beta = 256$) (see equation (21) and interpolation formula (22)) constancy rules. For each color component and image, the value printed in bold typeface highlights the best result.

3. Demosaicing Schemes

In this section, the main demosaicing schemes proposed in the literature are described. We distinguish two main procedures families, according to whether they scan the image plane or chiefly use the frequency domain.

3.1. Edge-adaptive Demosaicing Methods

Estimating the green plane before R and B ones is mainly motivated by the double amount of G samples in the CFA image. A fully populated G component plane will subsequently make the R and B plane estimation more accurate. As a consequence, the G component estimation quality becomes critical in the overall demosaicing performance, since any error in the G plane estimation is propagated in the following chrominance estimation step. Important efforts are therefore devoted to improve the estimation quality of the green component plane – usually assimilated to luminance –, especially in high-frequency areas. Practically, when the considered pixel lies on an edge between two homogeneous areas, missing components should be estimated along the edge rather than across it. In other words, neighboring pixels to be taken into account for interpolation should not belong to distinct objects. When exploiting the spatial correlation, a key issue is to determine the edge direction from CFA samples. As demosaicing methods presented in the following text generally use specific directions and neighborhoods in the image plane, some useful notations are introduced in figure 17.

3.1.1. Gradient-based Methods

Gradient computation is a general solution to edge direction selection. Hibbard's method (1995) uses horizontal and vertical gradients, computed at each pixel where the G component has to be estimated, in order to select the direction which provides the best green level estimation. Let us consider the $\{GRG\}$ CFA structure for instance (see figure 10a). Estimating the green level $\hat{G}$ at the center pixel is achieved in two successive steps :

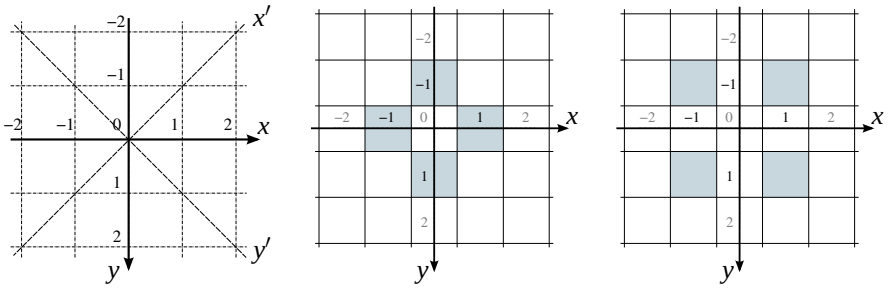
1. Approximate the gradient module (hereafter simply referred to as *gradient* for simplicity) according to horizontal and vertical directions, as :

$$\Delta^x = |G_{-1,0} - G_{1,0}|, \tag{25}$$
$$\Delta^y = |G_{0,-1} - G_{0,1}|. \tag{26}$$

2. Interpolate the green level as :

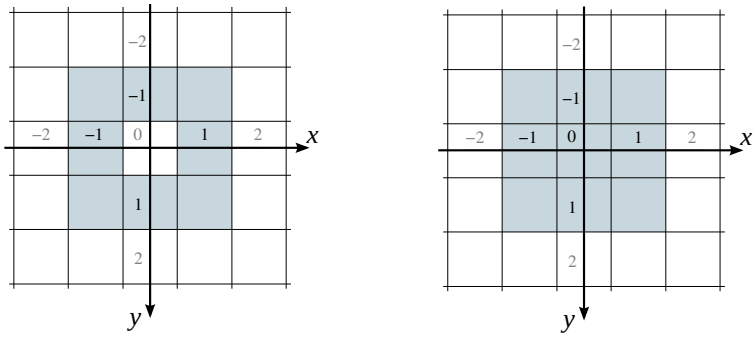
$$\hat{G} = \begin{cases} (G_{-1,0} + G_{1,0})/2 & \text{if } \Delta^x < \Delta^y, \quad (27a) \\ (G_{0,-1} + G_{0,1})/2 & \text{if } \Delta^x > \Delta^y, \quad (27b) \\ (G_{0,-1} + G_{-1,0} + G_{1,0} + G_{0,1})/4 & \text{if } \Delta^x = \Delta^y. \quad (27c) \end{cases}$$

Laroche and Prescott (1993) suggest to consider a 5×5 neighborhood for partial derivative approximations thanks to available surrounding levels, for instance $\Delta^x = |2R - R_{-2,0} - R_{2,0}|$. Moreover, Hamilton and Adams (1997) combine both approaches.



(a) Directions (b) N_4 neighborhood (c) N'_4 neighborhood

$$P(\delta x, \delta y) \in N_4 \quad \Leftrightarrow \quad (\delta x, \delta y) \in \{(0,-1)\,,\,(-1,0)\,,\,(1,0)\,,\,(0,1)\}$$
$$P(\delta x, \delta y) \in N'_4 \quad \Leftrightarrow \quad (\delta x, \delta y) \in \{(-1,-1)\,,\,(1,-1)\,,\,(-1,1)\,,\,(1,1)\}$$



(d) N_8 neighborhood ($N_8 \triangleq N_4 \cup N'_4$) (e) N_9 pixel set ($N_9 \triangleq N_8 \cup P(0,0)$)

FIG. 17: Notations for the main spatial directions and considered pixel neighborhoods.

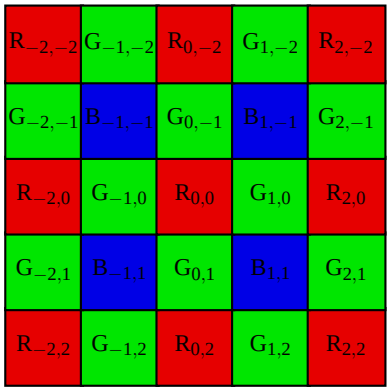


FIG. 18: 5×5 neighborhood with central $\{GRG\}$ structure in the CFA image.

To select the interpolation direction, these authors take into account both gradient and Laplacian second-order values by using the green levels available at nearby pixels and red (or blue) samples located 2 pixels apart. For instance, to estimate the green level at $\{GRG\}$ CFA structure (see figure 18), Hamilton and Adams use the following algorithm :

1. Approximate the horizontal Δ^x and vertical Δ^y gradients thanks to absolute differences as :

$$\Delta^x = |G_{-1,0} - G_{1,0}| + |2R - R_{-2,0} - R_{2,0}|, \tag{28}$$
$$\Delta^y = |G_{0,-1} - G_{0,1}| + |2R - R_{0,-2} - R_{0,2}|. \tag{29}$$

2. Interpolate the green level as :

$$\hat{G} = \begin{cases} (G_{-1,0} + G_{1,0})/2 + (2R - R_{-2,0} - R_{2,0})/4 & \text{if } \Delta^x < \Delta^y, \quad (30a) \\ (G_{0,-1} + G_{0,1})/2 + (2R - R_{0,-2} - R_{0,2})/4 & \text{if } \Delta^x > \Delta^y, \quad (30b) \\ (G_{0,-1} + G_{-1,0} + G_{1,0} + G_{0,1})/4 & \\ \quad + (4R - R_{0,-2} - R_{-2,0} - R_{2,0} - R_{0,2})/8 & \text{if } \Delta^x = \Delta^y. \quad (30c) \end{cases}$$

This proposal outperforms Hibbards' method. Indeed, precision is gained not only by combining two color component data in partial derivative approximations, but also by exploiting spectral correlation in the green plane estimation. It may be noticed that formula (30a) for the horizontal interpolation of green component may be split into one left $\hat{G}^g$ and one right $\hat{G}^d$ side parts :

$$\hat{G}^g = G_{-1,0} + (R - R_{-2,0})/2, \tag{31}$$
$$\hat{G}^d = G_{1,0} + (R - R_{2,0})/2, \tag{32}$$
$$\hat{G} = \left(\hat{G}^g + \hat{G}^d\right)/2. \tag{33}$$

Such interpolation is derived from the color difference constancy assumption, and hence exploits spectral correlation for green component estimation. Also notice that, in these equations, horizontal gradients are assumed to be similar for both red and blue components. A complete formulation has been given by Li and Randhawa (2005). As these authors show besides, the green component may more generally be estimated by a Taylor series as long as green levels are considered as a continuous function g which is differentiable in both main directions. The above equations (31) and (32) may then be seen as first-order approximations of this series. Indeed, in $\hat{G}^g$ case for instance, the horizontal approximation is written as $g(x) = g(x-1) + g'(x-1) \approx g(x-1) + (g(x) - g(x-2))/2$. Using the local constancy property of color component difference yields $\hat{G}_x - \hat{G}_{x-2} = R_x - R_{x-2}$, from which expression (31) is derived. Li and Randhawa suggest an approximation based on the second-order derivative, G^g estimation becoming :

$$\hat{G}^g = G_{-1,0} + (R - R_{-2,0})/2 + (R - R_{-2,0})/4 - (G_{-1,0} - G_{-3,0})/4, \tag{34}$$

for which a neighborhood size of 7×7 pixels is required. The additional term compared to (31) enables to refine the green component estimation. Similar reasoning may be used to select the interpolation direction. According to the authors, increasing the approximation order in such a way improves estimation results under the mean square error (*MSE*) criterion.

Another proposal comes from Su (2006), namely to interpolate the green level as a weighted sum of values defined by equations (30a) and (30b). Naming the latter respectively $\hat{G}^x = (G_{-1,0} + G_{1,0})/2 + (2R - R_{-2,0} - R_{2,0})/4$ and $\hat{G}^y = (G_{0,-1} + G_{0,1})/2 + (2R - R_{0,-2} - R_{0,2})/4$, horizontal and vertical interpolations are combined as :

$$\hat{G} = \begin{cases} w_1 \cdot \hat{G}^x + w_2 \cdot \hat{G}^y & \text{if } \Delta^x < \Delta^y, \quad (35a) \\ w_1 \cdot \hat{G}^y + w_2 \cdot \hat{G}^x & \text{if } \Delta^x > \Delta^y, \quad (35b) \end{cases}$$

where w_1 and w_2 are the weighting factors. Expression (30c) remains unchanged (i.e. $\hat{G} = \left(\hat{G}^x + \hat{G}^y\right)/2$ if $\Delta^x = \Delta^y$). The smallest level variation term must be weighted by the highest factor (i.e. $w_1 > w_2$) ; expressions (30a) and (30b) incidentally correspond to the special case $w_1 = 1$, $w_2 = 0$. Incorporating terms associated to high level variations allows to undertake high-frequency information in the green component interpolation expression itself. Su sets w_1 to 0.87 and w_2 to 0.13, since these weighting factor values yield the minimal average *MSE* (for the three color planes) over a large series of demosaiced images.

Other researchers, like Hirakawa and Parks (2005) or Menon et al. (2007), use the filterbank approach in order to estimate missing green levels, before selecting the horizontal or vertical interpolation direction at $\{GRG\}$ and $\{GBG\}$ CFA structures. This enables to design five-element mono-dimensional filters which are optimal towards criteria specifically designed to avoid interpolation artifacts. The proposed optimal filters (e.g. $h_{opt} = [-0.2569\ 0.4339\ 0.5138\ 0.4339\ -0.2569]$ for Hirakawa and Parks' scheme) are close to the formulation of Hamilton and Adams[2].

3.1.2. Component-consistent Demosaicing

Hamilton and Adam's method selects the interpolation direction on the basis of horizontal and vertical gradient approximations. But this may be inappropriate, and unsatisfying results may be obtained in areas with textures or thin objects. Figure 19 shows an example where horizontal Δ^x and vertical Δ^y gradient approximations do not allow to take the right decision for the interpolation direction. Wu and Zhang (2004) propose a more reliable way to select this direction, still by using a local neighborhood. Two candidate levels are computed to interpolate the missing green value at a given pixel : one using horizontal neighbors, the second using vertical neighboring pixels. Then, the missing R or B value is estimated in both horizontal and vertical directions with each of these G candidates. A final step consists in selecting the most appropriate interpolation direction, namely that minimizing the gradient sum on the color difference planes ($R - G$ and $B - G$) in the considered pixel neighborhood. This interpolation direction

[2]No detail will be here given about how R and B components are estimated by the above methods, for their originality mainly lies in the G component estimation.

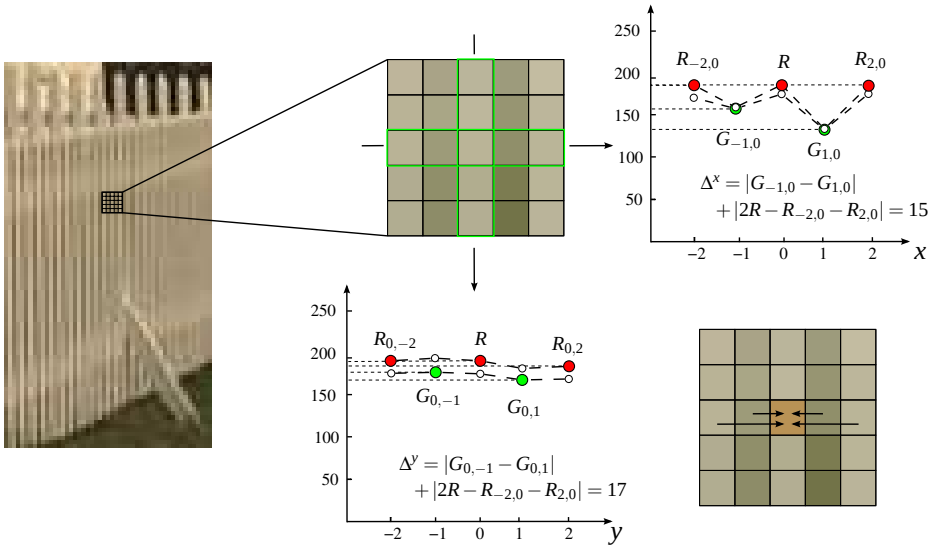


FIG. 19: Direction selection issue in Hamilton and Adams' interpolation scheme (1997), on an extract of the original image "Lighthouse" containing thin details. Plots highlight the R and G component values used for horizontal and vertical gradient computations : color dots represent available levels in the CFA image, whereas white dots are levels to be estimated. As $\Delta^x < \Delta^y$, horizontal neighboring pixels are wrongly used in $\hat{G}$ estimation. This is shown on the lower right subfigure, together with the erroneous demosaicing result (at center pixel only).

allows to select the levels – computed beforehand – to be taken into account for the missing component estimation.

More precisely, Wu and Zhang's approach proceeds in the following steps :

1. At each pixel where the green component is missing, compute two candidate levels : one denoted as $\hat{G}^x$ by using the horizontal direction (according to equation (30a)), and another $\hat{G}^y$ by using the vertical direction (according to (30b)). For other pixels, set $\hat{G}^x = \hat{G}^y = G$.
2. At each pixel where the green component is available, compute two candidate levels (one horizontal and one vertical) for each of the missing red and blue components. At $\{RGR\}$ CFA structure these levels are expressed as (see figure 10c) :

$$\hat{R}^x = G + \frac{1}{2}(R_{-1,0} - \hat{G}^x_{-1,0} + R_{1,0} - \hat{G}^x_{1,0}), \tag{36}$$

$$\hat{R}^y = G + \frac{1}{2}(R_{-1,0} - \hat{G}^y_{-1,0} + R_{1,0} - \hat{G}^y_{1,0}), \tag{37}$$

$$\hat{B}^x = G + \frac{1}{2}(B_{0,-1} - \hat{G}^x_{0,-1} + B_{0,1} - \hat{G}^x_{0,1}), \tag{38}$$

$$\hat{B}^y = G + \frac{1}{2}(B_{0,-1} - \hat{G}^y_{0,-1} + B_{0,1} - \hat{G}^y_{0,1}). \tag{39}$$

3. At each pixel with missing green component, compute two candidate levels for the missing chrominance component (i.e. $\hat{B}$ at R samples, and conversely). At $\{GRG\}$ CFA structure, the blue levels are estimated as (see figure 10a) :

$$\hat{B}^x = \hat{G}^x + \frac{1}{4}\sum_{P \in N'_4} (B(P) - \hat{G}^x(P)), \tag{40}$$

$$\hat{B}^y = \hat{G}^y + \frac{1}{4}\sum_{P \in N'_4} (B(P) - \hat{G}^y(P)), \tag{41}$$

   where N'_4 is composed of the four diagonal pixels (see figure 17c).
4. Achieve the final estimation at each pixel P by selecting one component triplet out of the two candidates computed beforehand in both horizontal and vertical directions. So as to use the direction for which variations of $(R-G)$ and $(B-G)$ component differences are minimal, the authors suggest the following selection criterion :

$$(\hat{R},\hat{G},\hat{B}) = \begin{cases} (\hat{R}^x,\hat{G}^x,\hat{B}^x) & \text{if } \Delta^x < \Delta^y, \quad (42a) \\ (\hat{R}^y,\hat{G}^y,\hat{B}^y) & \text{if } \Delta^x \geqslant \Delta^y, \quad (42b) \end{cases}$$

   where Δ^x and Δ^y are, respectively, the horizontal and vertical gradients on the difference plane of estimated colors. More precisely, these gradients are computed by considering all distinct (Q,Q') pixel pairs, respectively row-wise and column-wise, within the 3×3 window centered at P (see figure 17e) :

$$\Delta^x = \sum_{\substack{(Q,Q')\in N_9\times N_9 \\ y(Q)=y(Q')}} \left|\left(\hat{R}^x(Q)-\hat{G}^x(Q)\right)-\left(\hat{R}^x(Q')-\hat{G}^x(Q')\right)\right| + \left|\left(\hat{B}^x(Q)-\hat{G}^x(Q)\right)-\left(\hat{B}^x(Q')-\hat{G}^x(Q')\right)\right|, \quad (43)$$

$$\Delta^y = \sum_{\substack{(Q,Q')\in N_9\times N_9 \\ x(Q)=x(Q')}} \left|\left(\hat{R}^y(Q)-\hat{G}^y(Q)\right)-\left(\hat{R}^y(Q')-\hat{G}^y(Q')\right)\right| + \left|\left(\hat{B}^y(Q)-\hat{G}^y(Q)\right)-\left(\hat{B}^y(Q')-\hat{G}^y(Q')\right)\right|. \quad (44)$$

This method uses the same expressions as Hamilton and Adams' ones in order to estimate missing color components, but improves the interpolation direction decision by using a 3×3 window – rather than a single row or column – in which the gradient of color differences ($R-G$ and $B-G$) is evaluated so as to minimize its local variation.

Among other attempts to refine the interpolation direction selection, Hirakawa and Parks (2005) propose a selection criterion which uses the number of pixels with homogeneous colors in a local neighborhood. The authors compute the distances between the color point of the considered pixel and those of its neighbors in the CIE $L^*a^*b^*$ color space (defined in section 4.3.2), which better fits with the human perception of colors than *RGB* space. They design an homogeneity criterion with adaptive thresholding which reduces color artifacts due to incorrect selection of the interpolation direction. Chung and Chan (2006) nicely demonstrate that green plane interpolation is critical to the estimated image quality, and suggest to evaluate the local variance of color difference as an homogeneity criterion. The selected direction corresponds to minimal variance, which yields green component refinement especially in textured areas. Omer and Werman (2004) use a similar way to select the interpolation direction, except that the local color *ratio* variance is used. These authors also propose a criterion based on a local corner score. Under the assumption that demosaicing generates artificial corners in the estimated image, they apply the Harris corner detection filter (Harris and Stephens, 1988), and select the interpolation direction which provides the fewest detected corners.

3.1.3. Template Matching-based Methods

This family of methods aims at identifying a template-based feature in each pixel neighborhood, in order to interpolate according to the locally encountered feature. Such strategy has been first implemented by Cok in a patent dating back to 1986 (Cok, 1986)(Cok, 1994), in which the author classifies 3×3 neighborhoods into edge, stripe or corner features (see figure 20). The algorithm original part lies in the green component interpolation at each pixel P where it misses (i.e. at center pixel of $\{GRG\}$ or $\{GBG\}$ CFA structures) :

1. Compute the average green level available at the four nearest neighbor pixels of P (i.e. belonging to N_4, as defined on figure 17b). Examine whether each of these four green levels is lower (b), higher (h), or equal to their average. Sort these four

values in descending order, let $G_1 \geqslant G_2 \geqslant G_3 \geqslant G_4$, and compute their median $M = (G_2 + G_3)/2$.

2. Classify P neighborhood as :
   (a) *edge* if 3 h and 1 b are present, or 1 h and 3 b (see figure 20a) ;
   (b) *stripe* if 2 h and 2 b are present and opposite by pairs (see figure 20b) ;
   (c) *corner* if 2 h and 2 b are present and adjacent by pairs (see figure 20c).

   In the special case when two values are equal to the average, the encountered feature is taken as :
   (a) a stripe if the other two pixels b and h are opposite ;
   (b) an edge otherwise.
3. Interpolate the missing green level according to the previously identified feature :
   (a) for an edge, $\hat{G} = M$;
   (b) for a stripe, $\hat{G} = CLIP_{G_3}^{G_2}(M - (S - M))$, where S is the average green level over the eight neighboring pixels labeled as Q in figure 20d ;
   (c) for a corner, $\hat{G} = CLIP_{G_3}^{G_2}(M - (S' - M))$, where S' is the average green level over the four neighboring pixels labeled as Q in figure 20e, which are located on both sides of the borderline between b and h pixels.

   Function $CLIP_{G_3}^{G_2}$ simply limits the interpolated value to range $[G_3, G_2]$:

$$\forall \alpha \in \mathbb{R}, CLIP_{G_3}^{G_2}(\alpha) = \begin{cases} \alpha & \text{if } G_3 \leqslant \alpha \leqslant G_2, \\ G_2 & \text{if } \alpha > G_2, \\ G_3 & \text{if } \alpha < G_3. \end{cases} \tag{45}$$

This method, which classifies neighborhood features into three groups, encompasses three possible cases in an image. But the criterion used to distinguish the three features is still too simple, and comparing green levels with their average may not be sufficient to determine the existing feature adequately. Moreover, in case of a *stripe* feature, interpolation does not take into account this stripe direction.

Chang and Tan (2006) also implement a demosaicing method based on template-matching, but apply it on the color difference planes ($R - G$ and $B - G$) in order to interpolate R and B color components, G being estimated beforehand thanks to Hamilton and Adams' scheme described above. The underlying strategy consists in simultaneously exploiting the spatial and spectral correlations, and relies on a local edge information which causes fewer color artifacts than Cok's scheme. Although color difference planes carry less high-frequency information than color component planes (see figure 16), they can provide relevant edge information in areas with high spatial frequencies.

3.1.4. Adpative Weighted-Edge Method

Methods described above, as template-based or gradient-based ones, achieve interpolation according to the local context. They hence require prior neighborhood classification. The adaptive weighted-edge linear interpolation, first proposed by Kimmel

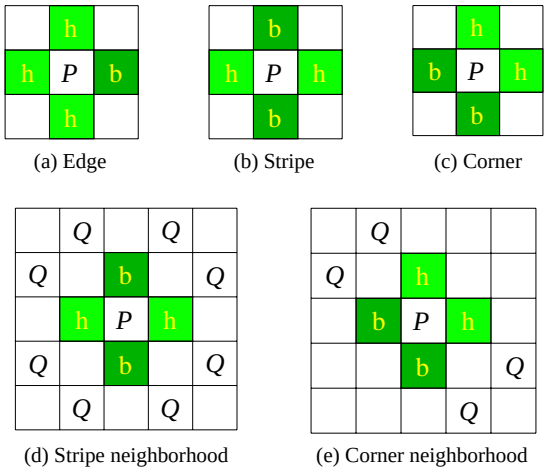


FIG. 20: Feature templates proposed by Cok to interpolate the green component at pixel P. These templates, which are defined modulo $\pi/2$, provide four possible *Edge* and *Corner* features, and two possible *Stripe* features.

(1999), is a method which merges these two steps into a single one. It consists in weighting each locally available level by a normalized factor as a function of a directional gradient. For instance, interpolating the green level at center pixel of $\{GRG\}$ or $\{GBG\}$ CFA structures is achieved as :

$$\hat{G} = \frac{w_{0,-1} \cdot G_{0,-1} + w_{-1,0} \cdot G_{-1,0} + w_{1,0} \cdot G_{1,0} + w_{0,1} \cdot G_{0,1}}{w_{0,-1} + w_{-1,0} + w_{1,0} + w_{0,1}}, \tag{46}$$

where $w_{\delta x,\delta y}$ coefficients are the weighting factors. In order to exploit spatial correlation, these weights are adjusted according to the locally encountered pattern.

Kimmel suggests to use local gradients to achieve weight computation. In a first step, directional gradients are approximated at a CFA image pixel P by using the levels of its neighbors. Gradients are respectively defined in horizontal, vertical, x'-diagonal (top-right to bottom-left) and y'-diagonal (top-right to bottom-right) directions (see figure 17a) over a 3×3 neighborhood by the following generic expressions :

$$\Delta^x(P) = (P_{1,0} - P_{-1,0})/2, \tag{47}$$
$$\Delta^y(P) = (P_{0,-1} - P_{0,1})/2, \tag{48}$$

$$\Delta^{x'}(P) = \begin{cases} \max\left(\left|(G_{1,-1} - G)/\sqrt{2}\right|, \left|(G_{-1,1} - G)/\sqrt{2}\right|\right) & \text{at } G \text{ locations}, \quad (49a) \\ (P_{1,-1} - P_{-1,1})/2\sqrt{2} & \text{elsewhere}, \quad (49b) \end{cases}$$

$$\Delta^{y'}(P) = \begin{cases} \max\left(\left|(G_{-1,-1} - G)/\sqrt{2}\right|, \left|(G_{1,1} - G)/\sqrt{2}\right|\right) & \text{at } G \text{ locations}, \quad (50a) \\ (P_{-1,-1} - P_{1,1})/2\sqrt{2} & \text{elsewhere}, \quad (50b) \end{cases}$$

where $P_{\delta x,\delta y}$ stands for the neighboring pixel of P, with relative coordinates $(\delta x,\delta y)$, in the CFA image. Here, R, G or B is not specified, since these generic expressions apply to all CFA image pixels, whatever the considered available component. However, we notice that all differences involved in equations (47) and (48) imply levels of a same color component.

The weight $w_{\delta x,\delta y}$ in direction d, $d \in \{x, y, x', y'\}$, is then computed from directional gradients as :

$$w_{\delta x,\delta y} = \frac{1}{\sqrt{1 + \Delta^d(P)^2 + \Delta^d(P_{\delta x,\delta y})^2}}, \quad (51)$$

where direction d used to compute the gradient Δ^d is defined by the center pixel P and its neighbor $P_{\delta x,\delta y}$. At the right-hand pixel $(\delta x,\delta y) = (1,0)$ as an example, the horizontal direction x is used for d ; $\Delta^d(P)$ and $\Delta^d(P_{1,0})$ are therefore both computed by expression (47) defining Δ^x, and the weight is expressed as :

$$w_{1,0} = \frac{1}{\sqrt{1 + (P_{-1,0} - P_{1,0})^2/4 + (P_{2,0} - P)^2/4}}. \quad (52)$$

Definition of weight $w_{\delta x,\delta y}$ is built so that a local transition in a given direction yields a high gradient value in the same direction. Consequently, weight $w_{\delta x,\delta y}$ is close to 0 for the neighbor $P_{\delta x,\delta y}$ and does not contribute much to the final estimated green level according to equation (46). On the opposite, weight $w_{\delta x,\delta y}$ is equal to 1 when the directional gradients are equal to 0.

Adjustments in weight w computation are proposed by Lu and Tan (2003), who use a Sobel filter to approximate the directional gradient, and the absolute – instead of square – value of gradients in order to boost computation speed. Such a strategy is also implemented by Lukac and Plataniotis (2005b).

Once then green plane has been fully populated thanks to equation (46), red and blue levels are estimated by using component ratios R/G and B/G among neighboring pixels. Interpolating the blue component is for instance achieved according to two steps (the red one being processed in a similar way) :

1. Interpolation at red locations (i.e. for $\{GRG\}$ CFA structure) :

$$\hat{B} = \hat{G} \cdot \frac{\sum_{P \in N_4'} w(P) \cdot \frac{B(P)}{\hat{G}(P)}}{\sum_{P \in N_4'} w(P)} = \hat{G} \cdot \frac{w_{-1,-1} \cdot \frac{B_{-1,-1}}{\hat{G}_{-1,-1}} + w_{1,-1} \cdot \frac{B_{1,-1}}{\hat{G}_{1,-1}} + w_{-1,1} \cdot \frac{B_{-1,1}}{\hat{G}_{-1,1}} + w_{1,1} \cdot \frac{B_{1,1}}{\hat{G}_{1,1}}}{w_{-1,-1} + w_{1,-1} + w_{-1,1} + w_{1,1}}. \quad (53)$$

2. Interpolation at other CFA locations with missing blue level (i.e. at $\{RGR\}$ and $\{BGB\}$ structures) :

$$\hat{B} = G \cdot \frac{\sum\limits_{P \in N_4} w(P) \cdot \frac{\hat{B}(P)}{\hat{G}(P)}}{\sum\limits_{P \in N_4} w(P)} = G \cdot \frac{w_{0,-1} \cdot \frac{\hat{B}_{0,-1}}{\hat{G}_{0,-1}} + w_{-1,0} \cdot \frac{\hat{B}_{-1,0}}{\hat{G}_{-1,0}} + w_{1,0} \cdot \frac{\hat{B}_{1,0}}{\hat{G}_{1,0}} + w_{0,1} \cdot \frac{\hat{B}_{0,1}}{\hat{G}_{0,1}}}{w_{0,-1} + w_{-1,0} + w_{1,0} + w_{0,1}}. \tag{54}$$

Once all missing levels have been estimated, Kimmel's algorithm (1999) achieves green plane refinement by using the color ratio constancy rule. This iterative refinement procedure is taken up by Muresan et al. (2000) with a slight modification : instead of using all N_8 neighboring pixels in step 1 below, only neighboring pixels with green available component are considered. The following steps describe this refinement scheme :

1. Correct the estimated green levels with the average of two estimations (one on the blue plane, the other on the red one), so that the constancy rule is locally enforced for color ratio G/R :

$$\hat{G} = \frac{1}{2}\left(\hat{G}^R + \hat{G}^B\right), \tag{55}$$

where :

$$\hat{G}^R \triangleq \widehat{R} \cdot \frac{\sum\limits_{P \in N_4} w(P) \cdot \frac{G(P)}{\widehat{R}(P)}}{\sum\limits_{P \in N_4} w(P)} \quad \text{and} \quad \hat{G}^B \triangleq \widehat{B} \cdot \frac{\sum\limits_{P \in N_4} w(P) \cdot \frac{G(P)}{\widehat{B}(P)}}{\sum\limits_{P \in N_4} w(P)},$$

$\widehat{B}$ and $\widehat{R}$ standing either for an estimated level or an available CFA value, according to the considered CFA structure ($\{GRG\}$ or $\{GBG\}$).

2. Correct then red and blue estimated levels at green locations, by using weighted R/G and B/G ratios at the eight neighboring pixels :

$$\hat{R} = G \cdot \frac{\sum\limits_{P \in N_8} w(P) \cdot \frac{\widehat{R}(P)}{\widehat{G}(P)}}{\sum\limits_{P \in N_8} w(P)} \tag{56}$$

and

$$\hat{B} = G \cdot \frac{\sum\limits_{P \in N_8} w(P) \cdot \frac{\widehat{B}(P)}{\widehat{G}(P)}}{\sum\limits_{P \in N_8} w(P)}. \tag{57}$$

3. Repeat the two previous steps twice.

This iterative correction procedure gradually enforces more and more homogeneous G/R and G/B color ratios, whereas the green component is estimated by using spectral correlation. Its convergence is however not always guaranteed, which may cause trouble for irrelevant estimated values. When a level occurring in any color ratio denominator is very close or equal to zero, the associated weight may not cancel the resulting bias. Figure 21c shows some color artifacts which are generated in this case. In pure yellow areas, quasi-zero blue levels cause a saturation of the estimated green component at R and B locations, which then alternate with original green levels.

Smith (2005) suggests to compute adaptive weights as $w_{\delta x,\delta y} = \frac{1}{1+4|\Delta^d(P)|+4|\Delta^d(P_{\delta x,\delta y})|}$, in order to reduce the division bias and contribution of pixels on both edge sides.

(a) Original image (b) Estimated image before correction (c) Estimated image after correction

FIG. 21: Demosaicing result achieved by Kimmel's method (1999), before and after the iterative correction steps. Generated artifacts are pointed out on image (c).

Lukac et al. (2006) choose to apply adaptive weighting on color *difference* planes for R and B component estimations, which avoids the above-mentioned artifacts during the iterative correction step. Tsai and Song (2007) take up the latter idea, but enhance the green plane interpolation procedure : weights are adapted to the local topology thanks to a preliminary distinction between homogeneous and edge areas.

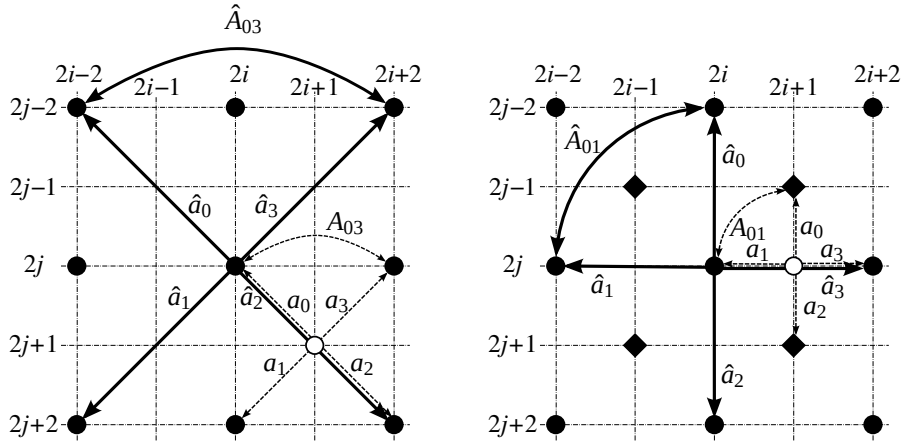
3.1.5. Local Covariance-based Methods

In his PhD dissertation, Li (2000) presents an interpolation scheme to increase the resolution of a gray-level image. Classical interpolation methods (bilinear and bicubic), based on spatial invariant models, tend to blur transitions and generate artifacts in high-frequency areas. Li's approach exploits spatial correlation by computing a local level covariance, without relying on directional gradients as do the above-mentioned methods in this section. Beyond resolution enhancement, the author applies this approach to demosaicing (Li and Orchard, 2001). In the CFA image, each of R, G or B color component plane may be viewed as a sub-sampled version of its respective, fully-populated estimated color plane. According to this consideration, a missing level in a given color plane is interpolated by using local covariance, preliminarily estimated from neighboring levels available in the same plane.

The underlying principle of this method may be better understood by considering the resolution enhancement problem first. More precisely, figure 22 illustrates how the resolution of a gray-level image can be doubled thanks to geometric duality, in a two-step procedure. The first step consists in interpolating $P_{2i+1,2j+1}$ level (represented by a white dot in figure 22a) from available $P_{2(i+k),2(j+l)}$ levels (black dots). The following linear combination of N_4' neighbors is used here :

$$\hat{P}_{2i+1,2j+1} = \sum_{k=0}^{1}\sum_{l=0}^{1} \alpha_{2k+l} P_{2(i+k),2(j+l)}, \tag{58}$$

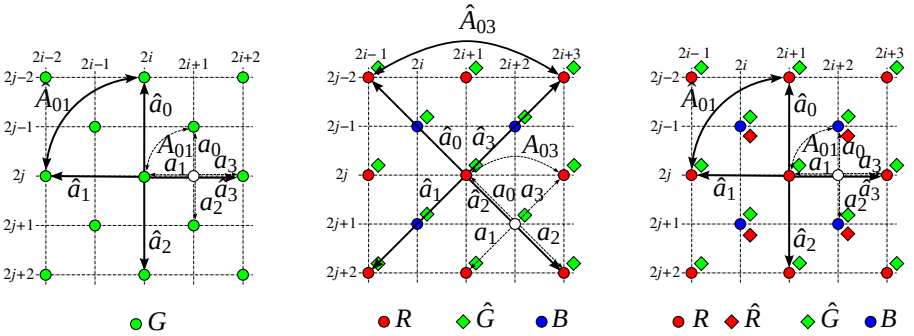
in which α_m coefficients, $0 \leqslant m \leqslant 3$, of $\vec{\alpha}$ are computed as follows (see justification



(a) Interpolating lattice $P_{2i+1,2j+1}$ from lattice $P_{2i,2j}$

(b) Interpolating lattice $P_{i,j}$ ($i+j$ odd) from lattice $P_{i,j}$ ($i+j$ even)

FIG. 22: Geometric duality between the low-resolution covariance and the high-resolution covariance. Black dots are the available levels at low resolution, and the white dot is the considered pixel to be interpolated. In subfigure (b), diamonds represent pixels estimated in the previous step.



(a) Interpolating G at R or B locations

(b) Interpolating R at B locations

(c) Interpolating R at G locations

FIG. 23: Geometric duality between covariances used in demosaicing. Color dots are the available components in the CFA image, and the white dot is the considered pixel to be interpolated. In subfigures (b) and (c), diamonds represent pixels estimated in the previous step, and spatial coordinates are shifted one pixel right.

and details in Li and Orchard, 2001) :

$$\vec{\alpha} = A^{-1}\vec{a}. \tag{59}$$

This expression incorporates the local covariance matrix $A \triangleq [A_{m,n}]$, $0 \leqslant m,n \leqslant 3$ between the four neighboring levels considered pair-wise (e.g. A_{03} in figure 22a), and the covariance vector $\vec{a} \triangleq [a_m]$, $0 \leqslant m \leqslant 3$ between the pixel level to be estimated and those of its four available neighbors (see figure 22a)[3]. The main issue is to get these covariances for the high-resolution image from levels which are available at low resolution. This is achievable by using the *geometric duality* principle : once covariance is computed in a local neighborhood of the low-resolution image, the equivalent covariance at high resolution is estimated by geometric duality which considers pixel pairs in the same direction at both resolutions. Under this duality principle, a_0 is for instance estimated by $\hat{a}_0$, A_{03} being replaced by $\hat{A}_{03}$ (see figure 22). The underlying assumption to approximate a_m by $\hat{a}_m$ and $A_{m,n}$ by $\hat{A}_{m,n}$, is that the local edge direction is invariant to image resolution.

The second step consists in estimating remaining unavailable levels, as for the white dot on figure 22b. Interpolation then relies on exactly the same principle as above, except that the available pixel lattice is now the previous one rotated by $\pi/4$.

Applying this method to demosaicing is rather straightforward :

1. Fill out the green plane at R and B locations by using :

$$\hat{G} = \sum_{P \in N_4} \alpha(P) G(P), \tag{60}$$

   where α coefficients are computed according to expression (59) and figure 23a.

2. Fill out the two other color planes, by exploiting the assumption of color difference ($R-G$ and $B-G$) constancy. For the red plane as example :

   (a) At B locations, interpolate the missing red level as :

$$\hat{R} = \hat{G} + \sum_{P \in N'_4} \alpha(P) \Big(R(P) - \hat{G}(P) \Big), \tag{61}$$

   where α coefficients are computed according to figure 23b.

   (b) At G locations, interpolate the missing red level as :

$$\hat{R} = G + \sum_{P \in N_4} \alpha(P) \Big(\widehat{R}(P) - \hat{G}(P) \Big), \tag{62}$$

   where α coefficients are computed according to figure 23c, $\widehat{R}$ being a value either available in I^{CFA} or estimated.

[3] Notations used here differ from those in the original publication (i.e. R and $\vec{r}$ for covariances) in order to avoid any confusion.

Although this method yields satisfying results (see next subsection), some limits may be pointed out. First, it requires the covariance matrix A to be invertible so that α coefficients can be computed. Li shows that this condition may not be verified in homogeneous areas of the image. Second, computing covariance matrices is a greedy processing task. To overcome those drawbacks, the author proposes a hybrid approach by using covariance-based interpolation only in edge areas, and a simple method (like bilinear interpolation) in homogeneous areas. This scheme avoids the covariance matrix invertibility issue, while decreasing computation time – since edge areas generally take up a small part of the whole image.

Leitão et al. (2003) observe that this method performs worse in textured areas than edge areas. They advise, for covariance estimation, to avoid considering pixels which are too far from the pixel to be interpolated. Asuni and Giachetti (2008) refine the detection scheme of areas in which the covariance estimation is appropriate for interpolation. These authors also improve the covariance matrix conditioning by adding a constant to pixel levels where they reach very low values. Tam et al. (2009) raise the covariance mismatch problem, which occurs when the geometric duality property is not satisfied, and solve it by extending the covariance matching into multiple directions. Multiple low-resolution training windows are considered, and the one that yields the highest covariance energy is retained to apply the linear interpolation according to generic equation (58). Lukin and Kubasov (2004) incorporate covariance-based interpolation for the green plane estimation, in a demosaicing algorithm combining several other techniques – notably Kimmel's. In addition, it is suggested to split non-homogeneous areas into textured and edge ones. The interpolation step is then achieved specifically to each kind of high-frequency contents.

3.1.6. Comparison Between Edge-adaptive Methods.

Finally, it is relevant to compare results achieved by the main exposed propositions which exploit spatial correlation. The key objective of these methods is to achieve the best estimation of green plane as possible, on which relies subsequent estimation of red and blue ones. Hence, we propose to examine the peak signal-to-noise ratio $PSNR^G$ (see expression (24)) of the estimated green plane, according to the experimental procedure described on figure 11. Table 2 shows the corresponding results, together with those achieved by bilinear interpolation for comparison. It can be noticed that all methods based on spatial correlation provide significant improvement in regard to bilinear interpolation. Among the six tested methods, Cok's (1986) and Li's (2001) estimate missing green levels by using only available green CFA samples, like bilinear interpolation ; all three generally provide the worst results. The green plane estimation may therefore be improved by using information from R and B components. In Kimmel's algorithm for instance (1999), green plane quality is noticeably enhanced, for 10 images out of 12, thanks to corrective iterations based on spectral correlation (see results of columns $Kimmel_0$ and $Kimmel_1$).

From these results may be asserted that any efficient demosaicing method should take advantage of both spatial and spectral correlations, simultaneously and for each color plane interpolation. Both methods proposed by Hamilton and Adams (1997) and by Wu and Zhang (2004) use the same expression to interpolate green levels, but different rules to select the interpolation direction. A comparison of respective results

Image	Bilinear	Hamilton	Kimmel_0	Kimmel_1	Wu	Cok	Li
1	38.982	44.451	40.932	28.244	**44.985**	39.320	39.999
2	32.129	37.179	33.991	37.947	**39.374**	32.984	34.305
3	37.477	43.161	39.870	38.207	**43.419**	38.161	38.780
4	28.279	34.360	31.643	34.673	**35.352**	30.420	30.705
5	36.709	42.603	39.291	41.477	**43.515**	38.103	38.849
6	33.168	38.148	34.913	38.659	**39.176**	33.762	34.354
7	35.682	40.650	37.605	40.978	**43.121**	36.734	38.356
8	32.804	39.434	36.261	39.514	**40.193**	35.073	35.747
9	35.477	40.544	37.470	39.603	**41.013**	36.219	36.656
10	32.512	37.367	34.224	**38.342**	38.125	33.117	36.656
11	34.308	38.979	35.934	38.321	**39.194**	34.837	35.107
12	30.251	34.451	31.248	35.145	**35.943**	30.150	30.173
Average	33.981	39.277	36.115	37.592	**40.284**	34.907	35.807

TAB. 2: Peak Signal-to-Noise Ratio (in decibels) of the green plane ($PSNR^G$), estimated by various interpolation methods. For each image, the best result is printed in bold typeface. Tested methods are here referred to chiefly by their first author's name : 1. Bilinear interpolation – 2. Hamilton and Adams' gradient-based method (1997) – 3 and 4. Kimmel's adaptive weighted-edge method (1999), before (Kimmel_0) and after (Kimmel_1) corrective iterations – 5. Wu and Zhang's component-consistent scheme (2004) – 6. Cok's method based on template matching (1986) – 7. Li's covariance-based method (2001).

show that careful selection of the interpolation direction is important for overall performance. This is all the most noticeable that, compared to other algorithms, computation complexity is rather low for both Hamilton and Adams' and Wu and Zhang's methods. Indeed, they do not require any corrective iteration step nor covariance matrix estimation step, which are computation-expensive operations.

3.2. Estimated Color Correction

Once the two missing components have been estimated at each pixel, a post-processing step of color correction is often applied to remove artifacts in the demosaiced image. To remove false colors in particular, a classical approach consists in strengthening spectral correlation between the three estimated color components. Such a goal may be reached first by median filtering, as described below. An iterative update of initial interpolated colors is also sometimes achieved, as Kimmel's corrective step (1999) presented in subsection 3.1.4. A still more sophisticated algorithm proposed by Gunturk et al. (2002) is described in detail in the second part of this section. Among other correction techniques of estimated colors, Li (2005) builds a demosaicing scheme by using a iterative approximation strategy with a spatially-adaptive stopping criterion ; he also studies the influence of the number of corrective iteration steps on the estimated image quality. Let us also mention here regularization schemes based on the Bayesian framework, as Markov Random Fields (see e.g. Mukherjee et al., 2001), which are however poorly adapted to real-time implementation.

3.2.1. Median Filtering

One of the most widespread techniques in demosaiced image post-processing is median filtering. Such a filter has been used for years to remove impulse noise in gray-level images, but also efficiently removes color artifacts without damaging local color variations. Freeman (1988) was the first person to take advantage of the median filter to remove demosaicing artifacts. Applied to the estimated planes of color differences $R-G$ and $B-G$, this filter noticeably improves the estimation provided by bilinear interpolation. As shown on figure 16d, these planes contain little high-frequency information. False estimated colors, which result from inconsistency between the local interpolation and those achieved in a neighborhood, may hence be more efficiently corrected on these planes while preserving object edges.

Median filtering is implemented in several works of the demosaicing literature. For instance, Hirakawa and Parks (2005) propose to iterate the following correction – without giving more details about the number of iteration steps nor the filter kernel size –, defined at each pixel as :

$$\hat{R}' = \hat{G} + M^{RG}, \tag{63}$$

$$\hat{G}' = \frac{1}{2}\left(\hat{R} + M^{GR} + \hat{B} + M^{GB}\right), \tag{64}$$

$$\hat{B}' = \hat{G} + M^{BG}, \tag{65}$$

where $\hat{R}'$, $\hat{G}'$ and $\hat{B}'$ denote the filtered estimated components, and $M^{kk'}$ is the output value of the median filter applied on estimated planes of color differences $\hat{I}^k - \hat{I}^{k'}$, $(k,k') \in \{R,G,B\}^2$. Lu and Tan (2003) use a slight variant of the latter, but advise to apply it selectively, since median filtering tends to attenuate color saturation in the estimated image. An appropriate strategy is proposed for the pre-detection of artifact-prone areas, where median filtering is then solely applied. However, Chang and Tan (2006) notice that median filtering applied to color *difference* planes, which still bear some textures around edges, tends to induce "zipper" artifact in these areas. In order to avoid filtering across edges in the color difference planes, edge areas are preliminarily detected thanks to a Laplacian filter.

Some artifacts may however remain in the median filtered image, which is mainly due to separate filtering of color difference planes ($R-G$ and $B-G$). An alternative may be to apply a vector median filter on the estimated color image while exploiting spectral correlation. The local output of such a filter is the color vector which minimizes the sum of distances to all other color vectors in the considered neighborhood. But according to Lu and Tan (2003), the vector filter brings out little superiority – if any – in artifact removal, compared with the median filter applied to each color difference plane. The authors' justification is that the estimation errors may be considered as additive noise which corrupts each color plane. These noise vector components are loosely correlated. In such conditions, Astola et al. (1990) show that vector median filtering does not achieve better results than marginal filtering on the color difference planes.

3.2.2. Alternating Projection Method

As previously mentioned in section 2.2.3, pixel levels bear strong spectral correlation in high spatial frequency areas of a natural color image. From this observation, Gunturk et al. (2002) aim at increasing the correlation of high-frequency information between estimated $\hat{R}$, $\hat{G}$ and $\hat{B}$ component planes, while keeping the CFA image data. These two objectives are enforced by using two convex constraint sets, on which the algorithm alternately projects estimated data. The first set is named "Observation" and ensures that interpolated data are consistent with those available in the CFA image. The second set, named "Detail", is based on a decomposition of each R, G and B plane into four frequency subbands thanks to a *filterbank* approach. A filterbank is a set of pass-band filters which decompose (*analyze*) the input signal into several subbands, each one carrying the original signal information in a particular frequency subband. On the opposite, a signal may be reconstructed (*synthesized*) in a filterbank by recombination of its subbands.

The algorithm uses an initially estimated image as starting point ; it may hence be considered as a – sophisticated – refinement scheme. To get the initial estimation $\hat{\mathbf{I}}_0$, any demosaicing method is suitable. The authors suggest to use Hamilton and Adams' scheme to estimate the green plane $\hat{I}_0^G$, and a bilinear interpolation to get the red $\hat{I}_0^R$ and blue $\hat{I}_0^B$ planes. Two main steps are achieved then, as illustrated on figure 24a :

1. Update the green plane by exploiting high-frequency information of red and blue planes. This enhances the initial green component estimation.

   (a) Use available red levels of the CFA image (or $\hat{I}_0^R$) to form a downsampled plane I_0^R of size $X/2 \times Y/2$, as illustrated on figure 24b.

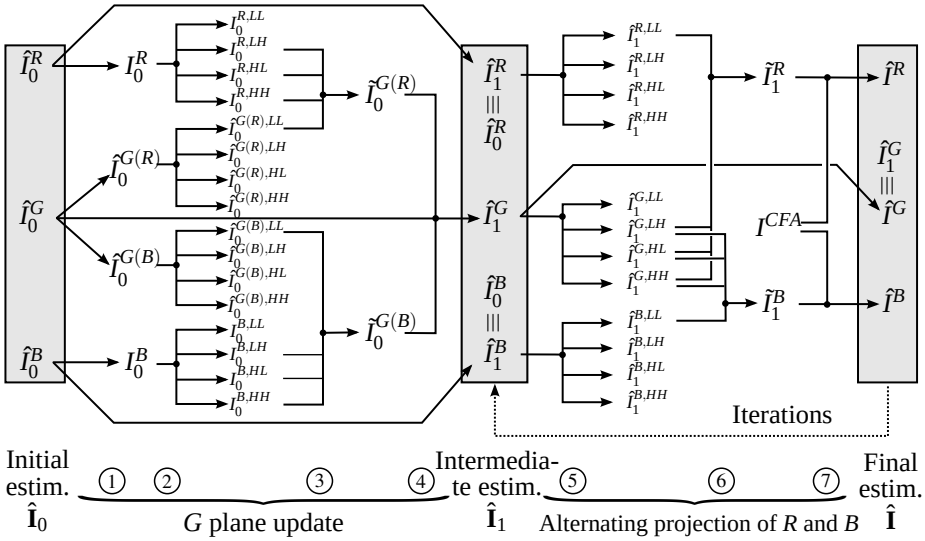
   (b) Sample, at the same R locations, green levels from the initial estimation $\hat{I}_0^G$ to form a downsampled plane $\hat{I}_0^{G(R)}$, also of size $X/2 \times Y/2$.

   (c) Decompose the downsampled plane I_0^R into four subbands :

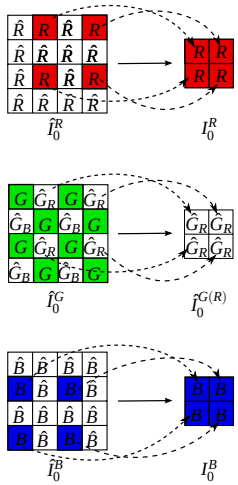
$$I_0^{R,LL}(x,y) = h_0(x) * \left[h_0(y) * I_0^R(x,y)\right], \quad (66)$$
$$I_0^{R,LH}(x,y) = h_0(x) * \left[h_1(y) * I_0^R(x,y)\right], \quad (67)$$
$$I_0^{R,HL}(x,y) = h_1(x) * \left[h_0(y) * I_0^R(x,y)\right], \quad (68)$$
$$I_0^{R,HH}(x,y) = h_1(x) * \left[h_1(y) * I_0^R(x,y)\right], \quad (69)$$

   and do the same with plane $\hat{I}_0^{G(R)}$. In their proposition, Gunturk *et al.* use a low-pass filter $H_0(z)$ and a high-pass filter $H_1(z)$ to analyze each plane respectively in low and high frequencies, as described above in subsection 2.2.1.

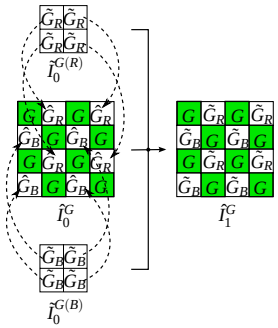
   (d) Use the low-frequency subband (LL) of $\hat{I}_0^{G(R)}$ and the three subbands of I_0^R with high frequencies (LH, HL and HH) to synthesize a re-estimated



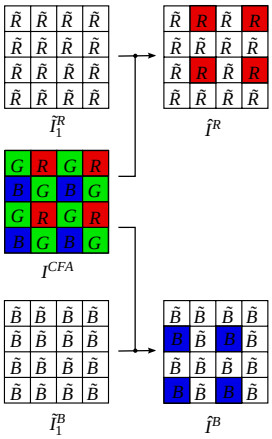
(a) Procedure outline. G channel update : ① Extraction of downsampled $X/2 \times Y/2$ planes (see details on figure 24b) – ② Subband analysis of each downsampled plane – ③ Synthesis of re-estimated downsampled green planes $\tilde{I}_0^{G(R)}$ and $\tilde{I}_0^{G(B)}$ at R and B locations – ④ Insertion of these planes into $\hat{I}_0^G$ (see details on figure 24c). Alternating projection of R and B components : ⑤ Subband analysis of intermediate estimation $\hat{\mathbf{I}}_1$ planes – ⑥ Synthesis of re-estimated red and blue planes – ⑦ Projection of these planes onto the "Observation" constraint set (see details on figure 24d).



(b) Extraction of downsampled planes from initial estimation.



(c) Insertion of re-estimated downsampled green planes into $\hat{I}_0^G$.



(d) Projection of re-estimated red and blue planes onto the "Observation" set.

FIG. 24: Demosaicing procedure proposed by Gunturk et al. (2002) from an initial estimation $\hat{\mathbf{I}}_0$.

downsampled green plane $\tilde{I}_0^{G(R)}$:

$$\begin{aligned}\tilde{I}_0^{G(R)}(x,y) &= g_0(x) * \left[g_0(y) * \hat{I}_0^{G(R),LL}(x,y)\right] + g_0(x) * \left[g_1(y) * I_0^{R,LH}(x,y)\right] \\ &\quad + g_1(x) * \left[g_0(y) * I_0^{R,HL}(x,y)\right] + g_1(x) * \left[g_1(y) * I_0^{R,HH}(x,y)\right].\end{aligned} \tag{70}$$

Filters $G_1(z)$ and $G_0(z)$ used for this synthesis have impulse responses $g_1 = [1\ 2\ -6\ 2\ 1]/8$ and $g_0 = [-1\ 2\ 6\ 2\ -1]/8$, respectively.

(e) Apply above instructions (a)-(d) similarly on the blue plane $\hat{I}_0^B$, which yields a second re-estimated downsampled green plane $\tilde{I}_0^{G(B)}$.

(f) Insert these two re-estimated downsampled estimations of the green plane at their respective locations in plane $\hat{I}_0^G$ (i.e. $\tilde{I}_0^{G(R)}$ at R locations, and $\tilde{I}_0^{G(B)}$ at B locations, as illustrated on figure 24c). A new full-resolution green plane $\hat{I}_1^G$ is obtained, which forms an intermediate estimated color image $\hat{\mathbf{I}}_1$ together with planes $\hat{I}_0^R$ and $\hat{I}_0^B$ from the initial estimation.

2. Update red and blue planes by alternating projections.

(a) Projection onto the "Detail" set : this step insures that high-frequency information is consistent between the three color planes, while preserving as much details as possible in the green plane. To achieve this, *a*) analyze the three color planes $\hat{I}_1^R$, $\hat{I}_1^G$ and $\hat{I}_1^B$ of the intermediate image $\hat{\mathbf{I}}_1$ into four subbands by using the same filterbank as previously (composed of $H_0(z)$ and $H_1(z)$) ; *b*) use the low-frequency subband of the red plane and the three high-frequency subbands of the green plane to synthesize a re-estimated red plane $\tilde{I}_1^R$, similarly to equation (70). At last, *c*) repeat the same operations on the blue plane to estimate $\tilde{I}_1^B$.

(b) Projection onto the "Observation" set : this step insures that estimated values are consistent with the ones available ("observed") in the CFA. The latter are simply inserted in re-estimated planes $\tilde{I}_1^R$ and $\tilde{I}_1^B$ at corresponding locations, as illustrated on figure 24d.

(c) Repeat above instructions (a) and (b) several times (the authors suggest to use eight iterations).

In short, high-frequency subbands at red and blue CFA locations are used first to refine the initial estimation of green color plane. The high-frequency information of red and blue planes is then determined by using green plane details so as to remove color artifacts. This method achieves excellent results, and is often considered as a reference in demosaicing benchmarks. However, its computation cost is rather high, and its performance depends on the quality of initial estimation $\hat{\mathbf{I}}_0$. A non-iterative implementation of this algorithm has been recently proposed (Lu et al., 2009), which achieves the same results as alternating projection at convergence, but at about height times faster speed.

Chen et al. (2008) exploit both subband channel decomposition and median filtering : a median filter is applied on the difference planes $\hat{I}^{R,LL} - \hat{I}^{G,LL}$ and $\hat{I}^{B,LL} - \hat{I}^{G,LL}$

of low-frequency subbands. Components are updated thanks to formulas proposed by Hirakawa and Parks (see equations (63) to (65)), but on each low-frequency subband. High-frequency subbands are not filtered, in order to preserve spectral correlation. The final estimated image is synthesized from the four frequency subbands, as in the alternating projection scheme of Gunturk *et al.*. Compared to the latter, median filtering mainly improves the demosaicing result on chrominance planes. Menon et al. (2006) notice that Gunturk *et al.*'s method tends to generate zipper effect along object boundaries. To avoid such artifact, a corrective technique is proposed, which uses the same subband decomposition principle but pre-determines the local edge direction (horizontal or vertical) on the estimated green plane. The authors suggest to use this particular direction to correct green levels by replacing high-frequency components with those of the available component (R or B) at the considered pixel. As the same direction is used to correct estimated $\hat{R}$ and $\hat{B}$ levels at G locations on the color difference planes, this technique insures interpolation direction consistency between color components, which has been shown to be important in subsection 3.1.2.

3.3. Demosaicing using the Frequency Domain

Some recent demosaicing schemes rely on a frequency analysis, by following an approach originated by Alleysson et al. (2005). The fundamental principle is to use a frequency representation of the Bayer CFA image[4]. In the spatial frequency domain, such a CFA image may be represented as a combination of a luminance signal and two chrominance signals, all three being well localized. Appropriate frequency selection therefore allows to estimate each of these signals, from which the demosaiced image can be retrieved. Notice that frequency-based approaches do not use Bayer's assumption that assimilates green levels to luminance, and blue and red levels to chrominance components.

3.3.1. Frequency Selection Demosaicing

A simplified derivation of Alleysson *et al.*'s approach has been proposed by Dubois (2005), whose formalism is retained here to present the general framework of frequency-domain representation of CFA images. Let us assume that, for each component k of a color image, $k \in \{R,G,B\}$, there exists an underlying signal f^k. Demosaicing then consists in computing an estimation $\hat{f}^k$ (coinciding with $\hat{I}^k$) at each pixel. Let us assume similarly that there exists a signal f^{CFA} which underlies the CFA image. This signal is referred to as *CFA signal* and coincides with I^{CFA} at each pixel. The CFA signal value at each pixel with coordinates (x,y) may be expressed as the sum of spatially sampled f^k signals :

$$f^{CFA}(x,y) = \sum_{k=R,G,B} f^k(x,y)m^k(x,y) \,, \tag{71}$$

[4]Let us make here clear that frequency (i.e. *spatial* frequency), expressed in cycles per pixel, corresponds to the inverse number of adjacent pixels representing a given level series according to a particular direction in the image (classically , the horizontal or vertical direction).

where $m^k(x,y)$ is the sampling function for the color component k, $k \in \{R,G,B\}$. For the Bayer CFA of figure 9, this set of functions is defined as :

$$m^R(x,y) = \frac{1}{4}\left(1-(-1)^x\right)\left(1+(-1)^y\right), \quad (72)$$

$$m^G(x,y) = \frac{1}{2}\left(1+(-1)^{x+y}\right), \quad (73)$$

$$m^B(x,y) = \frac{1}{4}\left(1+(-1)^x\right)\left(1-(-1)^y\right). \quad (74)$$

With the definition $\begin{bmatrix} f^L \\ f^{C1} \\ f^{C2} \end{bmatrix} \triangleq \begin{bmatrix} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ -\frac{1}{4} & \frac{1}{2} & -\frac{1}{4} \\ -\frac{1}{4} & 0 & \frac{1}{4} \end{bmatrix} \begin{bmatrix} f^R \\ f^G \\ f^B \end{bmatrix}$, the expression of f^{CFA} becomes :

$$\begin{aligned} f^{CFA}(x,y) &= f^L(x,y) + f^{C1}(x,y)(-1)^{x+y} + f^{C2}(x,y)\left((-1)^x - (-1)^y\right) \\ &= f^L(x,y) + f^{C1}(x,y)e^{j2\pi(x+y)/2} + f^{C2}(x,y)\left(e^{j2\pi x/2} - e^{j2\pi y/2}\right). \end{aligned} \quad (75)$$

The CFA signal may therefore be interpreted as the sum of a luminance component f^L at baseband, a chrominance component f^{C1} modulated at spatial frequency (horizontal and vertical) $(0.5,0.5)$, and of another chrominance component f^{C2} modulated at spatial frequencies $(0.5,0)$ and $(0,0.5)$. Such interpretation may be easily checked on an achromatic image, in which $f^R = f^G = f^B$: the two chrominance components are then equal to zero.

Provided that functions f^L, f^{C1} and f^{C2} can be estimated at each pixel from the CFA signal, estimated color levels $\hat{f}^R$, $\hat{f}^G$ and $\hat{f}^B$ are simply retrieved as :

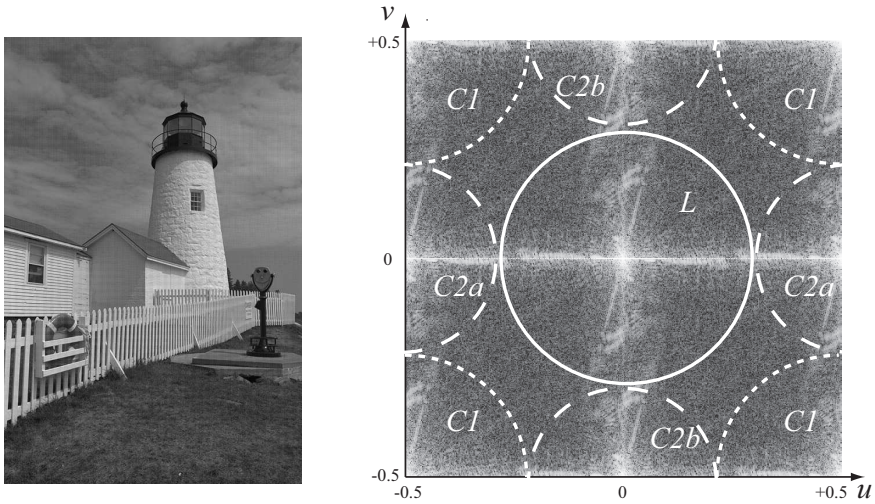
$$\begin{bmatrix} \hat{f}^R \\ \hat{f}^G \\ \hat{f}^B \end{bmatrix} = \begin{bmatrix} 1 & -1 & -2 \\ 1 & 1 & 0 \\ 1 & -1 & 2 \end{bmatrix} \begin{bmatrix} \hat{f}^L \\ \hat{f}^{C1} \\ \hat{f}^{C2} \end{bmatrix}. \quad (76)$$

To achieve this, the authors take the Fourier transform of the CFA signal (75) :

$$F^{CFA}(u,v) = F^L(u,v) + F^{C1}(u-0.5,v-0.5) + F^{C2}(u-0.5,v) - F^{C2}(u,v-0.5), \quad (77)$$

expression in which terms are, respectively, the Fourier transforms of $f^L(x,y)$, of $f^{C1}(x,y)(-1)^{x+y}$, and of the two signals defined as $f^{C2a}(x,y) \triangleq f^{C2}(x,y)(-1)^x$ and $f^{C2b}(x,y) \triangleq -f^{C2}(x,y)(-1)^y$.

It turns out that the energy of a CFA image is concentrated in nine zones of the frequency domain (see example of figure 25), centered on spatial frequencies according to equation (77) : energy of luminance $F^L(u,v)$ is mainly concentrated at the center of this domain (i.e. at low frequencies), whereas that of chrominance is located on its border (i.e. at high frequencies). More precisely, the energy of $F^{C1}(u-0.5,v-0.5)$ is located around diagonal zones ("corners" of the domain), that of $F^{C2}(u-0.5,v)$ along



(a) "Lighthouse" CFA image

(b) Normalized energy (frequencies in cycles/pixel)

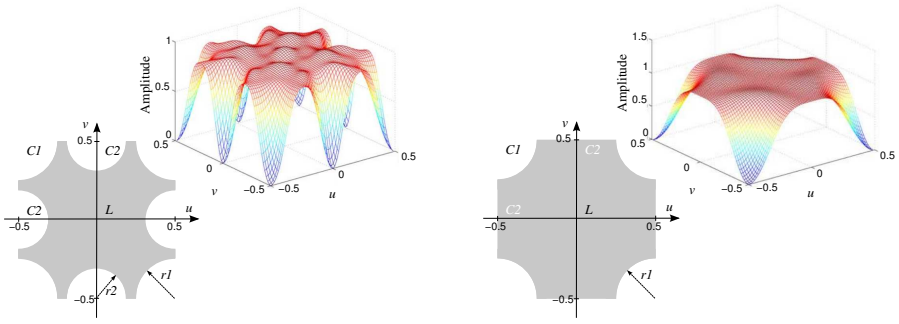
FIG. 25: Localization of the energy (Fourier transform module) of a CFA signal in the frequency domain (Alleysson et al., 2005).

u axis of horizontal frequencies, and that of $F^{C2}(u,v-0.5)$ along v axis of vertical frequencies. These zones are quite distinct, so that isolating the corresponding frequency components is possible by means of appropriately designed filters. But their bandwidth should be carefully selected, since the spectra of the three functions mutually overlap. In these frequency zones where luminance and chrominance cannot be properly separated, the *aliasing* phenomenon might occur and color artifacts be generated.

In order to design filter bandwidths which achieve the best possible separation of luminance (L) and chrominance ($C1$, $C2$), Dubois (2005) proposes an adaptive algorithm that mainly handles the spectral overlap between chrominance and high-frequency luminance components. The author observes that spectral overlap between luminance and chrominance chiefly occurs according to either the horizontal or the vertical axis. Hence he suggests to estimate f^{C2} by giving more weight to the sub-component of $C2$ ($C2a$ or $C2b$) that is least prone to spectral overlap with luminance. The implemented weight values are based on an estimation of the average directional energies, for which Gaussian filters (with standard deviation $\sigma = 3.5$ pixels and modulated at spatial frequencies $(0,0.375)$ and $(0.375,0)$ cycles per pixel) are applied to the CFA image.

3.3.2. Demosaicing by Joint Frequency and Spatial Analyses

Frequency selection is also a key feature used by Lian et al. (2007), who propose a hybrid method based on an analysis of both frequency and spatial domains. They state that the filter used by Alleysson *et al.* for luminance estimation may not be optimal. Moreover, since the parameters defining its bandwidth (see figure 26a) depend on the image content, they are difficult to be adjusted (Lian et al., 2005). Although low-pass filtering the CFA image allows to extract the luminance component, it removes



(a) Alleysson *et al.*

(b) Lian *et al.* (filter used at G locations)

FIG. 26: Filters (bandwidth and spectrum) used to estimate luminance, as proposed by Alleysson et al. (2005) and Lian et al. (2007).

the high-frequency information along horizontal and vertical directions. As the human eye is highly sensitive to the latter, such loss is prejudicial to the estimation quality. Lian *et al.* then notice that F^{C2} components in horizontal and vertical directions have same amplitudes but opposite signs[5]. Consequently, the luminance spectrum F^L at G locations is obtained as the CFA image spectrum from which $C1$ ("corner") component has been removed (see details in Lian et al., 2007). A low-pass filter is proposed to this purpose, which cancels $C1$ while preserving the high-frequency information along horizontal and vertical axes. This filter is inspired from Alleysson *et al.*'s, reproduced on figure 26a, but its bandpass is designed to remove $C1$ component only (see figure 26b). The main advantage of this approach is that luminance L spectrum bears less overlap with the spectrum of $C1$ than that of $C2$ (see example of figure 25b), which makes the filter design easier.

From these observations, Lian *et al.* propose a demosaicing scheme with three main steps (see figure 27) :

1. Estimate the luminance (denoted as $\hat{L}$) at G locations, by applying a low-pass filter on the CFA image to remove $C1$. Practically, the authors suggest to use the following 5×5 kernel, which gives very good results at low computational cost :

$$H = \frac{1}{64} \begin{bmatrix} 0 & 1 & -2 & 1 & 0 \\ 1 & -4 & 6 & -4 & 1 \\ -2 & 6 & 56 & 6 & -2 \\ 1 & -4 & 6 & -4 & 1 \\ 0 & 1 & -2 & 1 & 0 \end{bmatrix}. \tag{78}$$

2. Estimate the luminance at R and B locations by a spatial analysis. As isolating the spectrum of component $C2$ is rather difficult, the authors suggest an adaptive algorithm based on color difference constancy (exploiting spectral correlation) and adaptive weighted-edge linear interpolation (exploiting spatial correlation) :

[5] We keep here notations used by Alleysson *et al.* for $C1$ and $C2$, although switched by Lian *et al.*

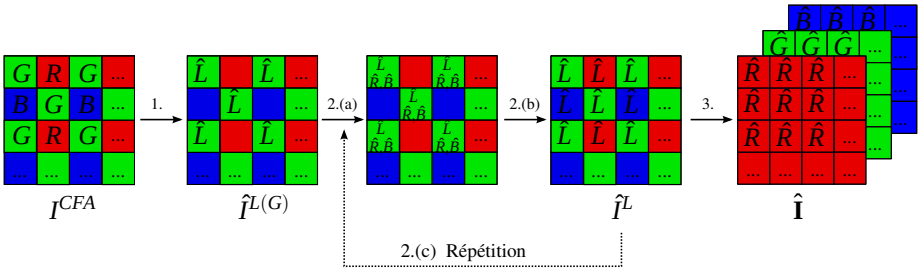


FIG. 27: Demosaicing scheme proposed by Lian et al. (2007) : 1. Luminance estimation at G locations – 2.(a) Pre-estimation of R and B components at G locations – 2.(b) Luminance estimation at R and B samples – 2.(c) Repetition of steps (a) and (b) – 3. Final color image estimation from the fully-populated luminance plane. Notation $\hat{I}^L$ used here for illustration sake coincides at each pixel with the luminance signal of expression (75), namely $\hat{L}_{x,y} \triangleq \hat{I}^L(x,y) \equiv \hat{f}^L(x,y)$.

(a) Pre-estimate R and B components at G locations, by simply averaging the levels of the two neighboring pixels at which the considered component is available.

(b) Estimate the luminance at R and B locations by applying, on the component difference plane $L-R$ or $L-B$, a weighted interpolation adapted to the local level transition. For instance, luminance $\hat{L}$ at R locations is estimated as follows :

$$\hat{L} = R + \frac{\sum\limits_{P \in N_4} w(P)\left(\hat{L}(P) - \hat{R}(P)\right)}{\sum\limits_{P \in N_4} w(P)}. \tag{79}$$

For the same $\{GRG\}$ CFA structure, weights $w(P) \equiv w_{\delta x,\delta y}$ are expressed by using the relative coordinates $P_{\delta x,\delta y}$ of the neighboring pixel as :

$$w_{\delta x,\delta y} = \frac{1}{1 + \left|R_{0,0} - R_{2\delta x,2\delta y}\right| + \left|\hat{L}_{\delta x,\delta y} - \hat{L}_{-\delta x,-\delta y}\right|}, \tag{80}$$

which achieves an adaptive weighted-edge interpolation, as in Kimmel's method (see section 3.1.4).

(c) Repeat the previous steps to refine the estimation : *a*) re-estimate R component (then B similarly) at G locations, by averaging $L-R$ levels at neighboring R locations ; *b*) re-estimate L at R (then B) locations according to equation (79) (weights $w(P)$ remaining unchanged).

3. From the fully-populated luminance plane $\hat{I}^L$, estimate the two missing components at each pixel of the CFA image by using bilinear interpolation :

$$\hat{I}^k_{x,y} = \hat{I}^L_{x,y} + \left(H^k * \varphi^k\left(I^{CFA} - \hat{I}^L\right)\right)(x,y), \tag{81}$$

where $\varphi^k(I)(x,y)$, $k \in \{R,G,B\}$ is the plane defined by expression (7) and shown on figure 12, and where convolution kernels H^k which achieve bilinear interpolation are defined by expressions (8) and (9)[6].

The above approach does not require to design specific filters in order to estimate $C1$ and $C2$ components, as do methods using the frequency domain only (Dubois uses for instance complementary asymmetric filters). Lian *et al.* show that their method globally outperforms other demosaicing schemes according to *MSE* (or *PSNR*) criterion. The key advantage seems to lie in exploiting the frequency domain at G locations only. According to results presented by Lian et al. (2007), luminance estimations are less error-prone than green level estimations provided by methods which chiefly scan the spatial image plane (shown in table 2).

3.4. Conclusion

An introduction to the demosaicing issue and to its major solutions has been exposed in the above section. After having described why such a processing task is required in mono-CCD color cameras, the various CFA solutions have been presented. Focusing on the Bayer CFA, we have detailed the formalism in use throughout the paper. The simple bilinear interpolation has allowed us to introduce both artifact generation that demosaicing method have to overcome, and two major rules widely used in the proposed approaches : spatial and spectral correlations.

The vast majority of demosaicing methods strive to estimate the green plane first, which bear the most high-frequency information. The quality of this estimation strongly influences that of red and blue planes. When exploiting spatial correlation, we experimentally show that a correct selection of the interpolation direction is crucial to reach a high interpolation quality for green levels. Moreover, component-consistent directions should be enforced in order to avoid color artifact generation. Spectral correlation is often taken into account by interpolating on the difference, rather than ratio, of component planes. An iterative post-processing step of color correction is often achieved, so as to improve the final result quality by reinforcing spectral correlation.

Demosaicing methods may exploit spatial and/or frequency domains. The spatial domain has been historically used first, and many studies are based on it. More recently, authors exploit the frequency domain, which opens large perspectives. Such approaches indeed allow to avoid using – at least partially or in a first step – the heuristic rule of color difference constancy to take spectral correlation into account. In all cases where such assumptions are not fulfilled, even locally, exploiting the frequency domain is an interesting solution. Dubois foresaw several years ago (2005) that frequency selection approaches are preeminently promising. This will be corroborated is the next sections, dedicated to the objective quality evaluation of images demosaiced by the numerous presented methods. Already mentioned criteria (*MSE* and *PSNR*) will be completed by measures suited to human color perception, and new specific ones dedicated to the local detection of demosaicing artifacts.

[6] Notice that $\varphi^k(I)$ may equally be expressed as $\varphi^k(I)(x,y) = I(x,y)m^k(x,y)$, where sampling functions m^k are defined by (72) to (74).

4. Objective Evaluation Criteria for Demosaiced Images

4.1. Introduction

The performances reached by different demosaicing schemes applied to the same CFA image can be very different. Indeed, different kinds of artifacts which alter the image quality, can be generated by demosaicing schemes. A description of these artifacts is given in subsection 4.2.

Measuring the performance reached by a demosaicing scheme requires to evaluate the quality of its output image. Indeed, such a measurement helps to compare the performances of the different schemes. For this purpose, we always follow the same experimental procedure (see figure 11). First, we simulate the color sampling by keeping only one out of the three color components at each pixel of the original image $\mathbf{I}$, according to the Bayer CFA mosaic. Then, we apply the considered demosaicing scheme to obtain the estimated color image $\hat{\mathbf{I}}$ (hereafter called *demosaiced image*) from the CFA samples. Finally, we measure the demosaicing quality by comparing the original and demosaiced images.

The main strategy of objective comparison is based on error estimation between the original and demosaiced images. In subsection 4.3, we present the most used criteria for objective evaluation of the demosaiced image.

The objective criteria are generally based on a pixel-wise comparison between the original and the estimated colors. These fidelity criteria are not specifically sensitive to one given artifact. Hence, in subsection 4.5, we present new measurements which quantify the occurrences of demosaicing artifacts.

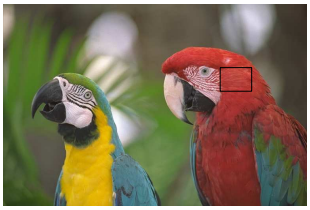
Since demosaicing methods intend to produce “perceptually satisfying” images, the most widely used evaluation criteria are based on the fidelity to the original images. Rather than displaying images, our goal is to apply automatic image analysis procedures to the demosaiced images in order to extract features. These extracted features are mostly derived from either colors or detected edges in the demosaiced images. Since the quality of features is sensitive to the presence of artifacts, we propose to quantify the demosaicing performance by measuring the rates of erroneously detected edge pixels. This evaluation scheme is presented in the last subsection.

4.2. Demosaicing Artifacts

The main artifacts caused by demosaicing are blurring, false colors and zipper effect. In this part, we present those artifacts on examples and explain their causes by considering the spatial and frequency domains.

4.2.1. Blurring Artifact

Blurring is located in areas where high frequency information, representing precise details or edges, is altered or erased. Figure 28, illustrates different blurring levels according to the used demosaicing scheme. A visual comparison between the original image 28b and image 28c which has been demosaiced by bilinear interpolation, shows that this scheme causes severe blurring. Indeed, some details of the parrot plumage are not retrieved by demosaicing and blurring is generated by low-pass filtering. As stated in section 2.2.1, this interpolation can be achieved by a convolution applied to



(a) Original Image.

(b) (c) (d)

FIG. 28: Blurring in the demosaiced image. Image (b) is an extract from the original image (a), located by a black box. Images (c) and (d) are the corresponding extracts of the images respectively demosaiced by bilinear interpolation and by Hamilton and Adams' (1997) schemes.

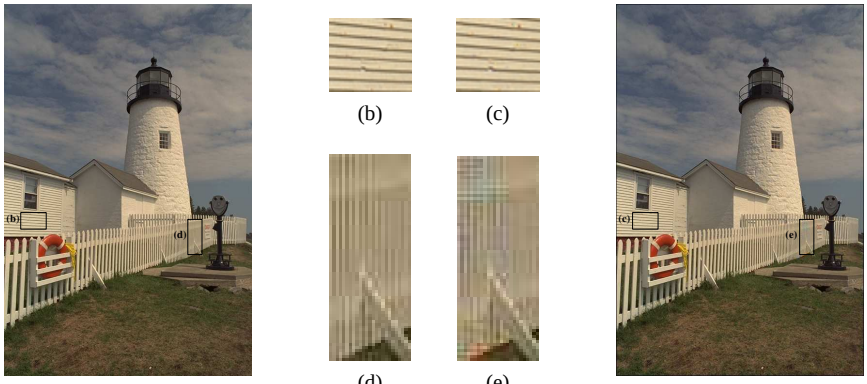
each sampled color component plane (see expression (10)). The corresponding filters, whose masks H^k are given by expressions (8) and (9), reduce high frequencies. Hence, fine details may be not properly estimated in the demosaiced image (see figure 28c).

This artifact is less visible in image 28d, which has been demosaiced by Hamilton and Adams' scheme (1997). A visual comparison with image 28c shows that this scheme, presented in section 3.1.1, generates a small amount of visible blurring. It first estimates vertical and horizontal gradients, then interpolates the green levels along the direction with the lowest gradient module, i.e. by using as homogeneous levels as possible. This selection of neighbors used to interpolate the missing green level at a given pixel, tends to avoid blurring.

4.2.2. Zipper Effect

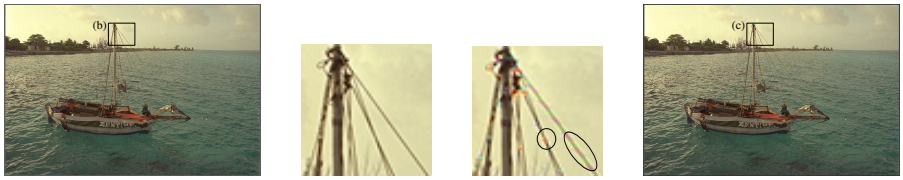
Let us examine figure 29, and more precisely images 29b and 29d which are extracted from the original "Lighthouse" image 29a. Images 29c and 29e are the corresponding extracts from the demosaicing result of Hamilton and Adams' scheme (1997). On image 29e, one can notice repetitive patterns in transition areas between homogeneous ones. This phenomenon is called *zipper effect*.

The main reason for this artifact is the interpolation of levels which belong to homogeneous areas representing different objects. It occurs at each pixel where the interpolation direction (horizontal or vertical) is close to that of the color gradient computed in the original image. Image 29c does not contain any zipper effect, since the interpolation direction is overall orthogonal to that of a color gradient, hence close to the transition direction between homogeneous areas. Oppositely, image 29e contains strong zipper effect. In this area with high spatial frequencies along the horizontal direction, the scheme often fails to determine the correct gradient direction (see section 3.1.2 and



(a) Original image (f) Demosaiced image

FIG. 29: Zipper effect due to erroneous selection of the interpolation direction. Images (b) and (d) are two extracts from the original image (a), located by black boxes. Images (c) and (e) are the corresponding extracts from the image (f) demosaiced by Hamilton and Adams's scheme (1997).



(a) Original image (b) (c) (d) Demosaiced image

FIG. 30: False colors on a diagonal detail. Image (b) is an extract from the original image (a), located by a black box. Image (c), on which artifacts are circled in black, is the corresponding extract from image (d) demosaiced by Hamilton and Adams's scheme (1997).

figure 19).

The other main reason is related to the arrangement, in the CFA image, of pixels whose green level is not available. Indeed, these pixels where the green levels can be erroneously estimated, are arranged in staggered locations.

4.2.3. False Colors

False color at a pixel corresponds to a large distance between the original color and the estimated one, in the acquisition color space *RGB*. Figures 30c and 31c show that this phenomenon is not characterized by a specific geometrical structure in the image.

Incorrect estimation of the color components may cause perceptible false colors, in particular in areas with high spatial frequencies.

(a) Original Image (b) (c)

FIG. 31: False colors generated on a textured area. (b) Extract from the original image (a), located by a black box. (c) Extract demosaiced by Wu and Zhang scheme (2004), with artifacts circled in black.

4.2.4. Artifacts Described in the Frequency Domain

The representation of the CFA color samples in the frequency domain, proposed by Alleysson et al. (2005), also allows to explain the reasons why artifacts are generated by demosaicing schemes.

As seen in section 3.3.1, the CFA image signal is made up of a luminance signal, mainly modulated at low spatial frequencies, and of two chrominance signals, mainly modulated at high frequencies (see figure 25 page 48). Therefore, demosaicing can be considered as an estimation of luminance and chrominance components. Several schemes which analyze the frequency domain (Alleysson et al., 2005; Dubois, 2005; Lian et al., 2007) estimate the missing levels by selective filters applied to the CFA image. The four possible artifacts caused by frequency analysis are shown in figure 32 extracted from (Alleysson et al., 2005) : excessive blurring, grid effect, false colors and watercolor. When the bandwidth of the filter applied to the CFA image to estimate the luminance is too narrow, an excessive blurring occurs in the demosaiced image (see figure 32a). When the bandwidth of this filter is too wide, it may select high frequencies in zones of chrominance. Such a case can result in a grid effect, especially visible in flat (homogeneous) areas of the image (see figure 32b). Moreover, false colors appear when the chrominance filters overlap with the luminance filter in the frequency domain (see figure 32c). Finally, when the chrominance filter is too narrow, watercolor effect may appear as colors which are “spread beyond” the edges of an object (see figure 32d).

These artifacts are caused by a bad conception of the selective filters used to estimate luminance and chrominance. They can also be generated by demosaicing methods which spatially scan the image. Indeed, several spatial demosaicing schemes generate blurring and false colors since they tend to under-estimate luminance and over-estimate chrominance. Kimmel’s (1999) and Gunturk et al.’s (2005) schemes also generate grid effect and watercolor.

4.3. Classical Objective Criteria

All the described artifacts are due to errors in color component estimation. The classical objective evaluation criteria sum up the errors between levels in the original and demosaiced images. At each pixel, the error between the original and demosaiced images is quantized thanks to a distance between two color points in a three-dimensional color space (Busin et al., 2008). In this subsection, we regroup the most

(a) Blurring (b) Grid effect

(c) False color (d) Watercolor

FIG. 32: Four kinds of artifacts caused by demosaicing (Alleysson et al., 2005).

widely used measurements into two categories, namely the fidelity and perceptual criteria.

4.3.1. Fidelity Criteria

These criteria use colors coded in the *RGB* acquisition color space in order to estimate the fidelity of the demosaiced image compared with the original image.

1. Mean Absolute Error.
   This criterion evaluates the mean absolute error between the original image $\mathbf{I}$ and the demosaiced image $\hat{\mathbf{I}}$. Denoted by *MAE*, it is expressed as (Chen et al., 2008; Li and Randhawa, 2005) :

$$MAE(\mathbf{I},\hat{\mathbf{I}}) = \frac{1}{3XY} \sum_{k=R,G,B} \sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} \left| I_{x,y}^{k} - \hat{I}_{x,y}^{k} \right|, \tag{82}$$

   where $I_{x,y}^{k}$ is the level of the color component k at the pixel whose spatial coordinates are (x,y) in the image $\mathbf{I}$. X and Y are respectively the number of columns and rows of the image.
   The *MAE* criterion can be used to measure the estimation errors of a specific color component. For example, this criterion is evaluated on the red color plane as :

$$MAE^{R}(\mathbf{I},\hat{\mathbf{I}}) = \frac{1}{XY} \sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} \left| I_{x,y}^{R} - \hat{I}_{x,y}^{R} \right|. \tag{83}$$

   MAE values range from 0 to 255, and the demosaicing quality is considered as better as its value is low.
2. Mean Square Error.
   This criterion measures the mean quadratic error between the original image and the demosaiced image. Denoted by *MSE*, it is defined as (Alleysson et al., 2005) :

$$MSE(\mathbf{I},\hat{\mathbf{I}}) = \frac{1}{3XY} \sum_{k=R,G,B} \sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} (I_{x,y}^{k} - \hat{I}_{x,y}^{k})^{2}. \tag{84}$$

   The *MSE* criterion can also measure the error on each color plane, as in equation (23). The optimal quality of demosaicing is reached when *MSE* is equal to 0, whereas the worst is measured when *MSE* is close to 255^2.
3. Peak Signal-to-Noise Ratio.
   The *PSNR* criterion is a widely used distortion measurement to estimate the quality of image compression. Many authors (e.g. Alleysson et al., 2005; Hirakawa and Parks, 2005; Lian et al., 2007; Wu and Zhang, 2004) use this criterion to quantify the performance reached by demosaicing schemes. The *PSNR* is expressed in decibels as :

$$PSNR(\mathbf{I},\hat{\mathbf{I}}) = 10 \cdot \log_{10} \left(\frac{d^2}{MSE(\mathbf{I},\hat{\mathbf{I}})} \right), \tag{85}$$

where d is the maximum color component level. When the color components are quantized with 8 bits, d is set to 255.

Like the preceding criteria, *PSNR* can be applied to a specific color plane. For the red color component, it is defined as :

$$PSNR^R(\mathbf{I},\hat{\mathbf{I}}) = 10 \cdot \log_{10}\left(\frac{d^2}{MSE^R(\mathbf{I},\hat{\mathbf{I}})}\right). \tag{86}$$

The higher the *PSNR* value is, the better is the demosaicing quality. The *PSNR* measured on demosaiced images generally ranges from 30 to 40 dB (i.e. *MSE* ranges from 65.03 to 6.50).

4. Correlation.

   A correlation measurement between the original image and the demosaiced image is used by Su and Willis (2003) to quantify the demosaicing performance. The correlation criterion between two gray-level images I and $\hat{I}$ is expressed as :

$$C(I,\hat{I}) = \left| \frac{\left(\sum_{x=0}^{X-1}\sum_{y=0}^{Y-1} I_{x,y}\hat{I}_{x,y}\right) - XY\mu\hat{\mu}}{\left[\left(\sum_{x=0}^{X-1}\sum_{y=0}^{Y-1} {I_{x,y}}^2\right) - XY\mu^2\right]^{1/2}\left[\left(\sum_{x=0}^{X-1}\sum_{y=0}^{Y-1} {\hat{I}_{x,y}}^2\right) - XY\hat{\mu}^2\right]^{1/2}} \right|, \tag{87}$$

   where μ and $\hat{\mu}$ are the mean gray levels in the two images.

   When a color demosaiced image is considered, one estimates the correlation level $C^k\left(I^k,\hat{I}^k\right)$, $k \in \{R,G,B\}$, between the original and demosaiced color planes. The mean of the three correlation levels is used to measure the quality of demosaicing. The correlation levels C range between 0 and 1, and a measurement close to 1 can be considered as a satisfying demosaicing quality.

4.3.2. Perceptual Criteria

The preceding criteria are not well consistent with quality estimation provided by the human visual system. That is the reason why new measurements have been defined, which operate in perceptually uniform color spaces (Chung and Chan, 2006).

1. Estimation Error in the CIE $L^*a^*b^*$ color space.

   The CIE $L^*a^*b^*$ color space is recommended by the International Commission on Illumination to measure the distance between two colors (Busin et al., 2008). This space is close to a perceptually uniform color space which has not been completely defined yet. So, the Euclidean distance in the CIE $L^*a^*b^*$ color space is a perceptual distance between two colors.

   The three color components (R,G,B) at a pixel are first transformed into (X,Y,Z) components according to a CIE XYZ linear operation. Then, the color components CIE $L^*a^*b^*$ are expressed as :

$$L^* = \begin{cases} 116 \times \sqrt[3]{Y/Y_n} - 16 & \text{if } Y/Y_n > 0.008856, \quad (88a) \\ 903.3 \times Y/Y_n & \text{otherwise,} \quad (88b) \end{cases}$$

$$a^* = 500 \times (f(X/X_n) - f(Y/Y_n)), \quad (89)$$
$$b^* = 200 \times (f(Y/Y_n) - f(Z/Z_n)), \quad (90)$$

with :

$$f(x) = \begin{cases} \sqrt[3]{x} & \text{if } Y/Y_n > 0.008856\,, \quad (91a) \\ 7.787x + \frac{16}{116} & \text{otherwise,} \quad (91b) \end{cases}$$

where the used reference white is characterized by the color components (X_n, Y_n, Z_n). We can notice that L^* represents the eye response to a specific luminance level, whereas a^* and b^* components correspond to chrominance. The component a^* represents an opposition of colors Red–Green, and b^* corresponds to an opposition of colors Blue–Yellow.

The color difference is defined as the distance between two color points in this color space. Then, the estimation error caused by demosaicing is the mean error processed with all image pixels :

$$\Delta E^{L^*a^*b^*}(\mathbf{I}, \hat{\mathbf{I}}) = \frac{1}{XY} \sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} \sqrt{\sum_{k=L^*,a^*,b^*} \left(I^k_{x,y} - \hat{I}^k_{x,y}\right)^2}\,. \quad (92)$$

The lower $\Delta E^{L^*a^*b^*}$ is, the lower is the perceptual difference between the original and demosaiced images, and the higher is the demosaicing quality.

2. Estimation Error in the S-CIE $L^*a^*b^*$ color space.

   In order to introduce spatial perception properties of the human visual system, Zhang and Wandell (1997) propose a new perceptual color space, called S-CIE $L^*a^*b^*$. The color components (R,G,B) are first transformed into the color space XYZ which does not depend on the acquisition device. Then, these color components are converted into the antagonist color space AC_1C_2, where A represents the perceived luminance and C_1, C_2, the chrominance information in terms of opposition of colors Red–Green and Blue–Yellow, respectively. The three component planes are then separately filtered by Gaussian filters with specific variances, which approximate the contrast sensitivity functions of the human visual system.

   The three filtered components A, C_1 and C_2 are converted back into (X,Y,Z) components, which are then transformed into CIE $L^*a^*b^*$ color space thanks to equations (88) and (89). Once the color components L^*, a^* and b^* have been computed, the estimation error ΔE in S-CIE $L^*a^*b^*$ is defined by equation (92). This measurement was used by Li (2005), Su (2006) and Hirakawa and Parks (2005) to measure the demosaicing quality.

3. Normalized Color Difference in the CIE $L^*u^*v^*$ color space.

   The CIE proposes another perceptually uniform color space called CIE $L^*u^*v^*$, whose luminance L^* is the same as that of CIE $L^*a^*b^*$ color space. The chrominance components are expressed as :

$$u^* = 13 \times L^* \times (u' - u'_n), \tag{93}$$
$$v^* = 13 \times L^* \times (v' - v'_n), \tag{94}$$

   with :

$$u' = \frac{4X}{X + 15Y + 3Z}\ , \tag{95}$$
$$v' = \frac{9Y}{X + 15Y + 3Z}\ , \tag{96}$$

   where u'_n et v'_n are the chrominance of the reference white.

   The criterion of normalized color difference *NCD* is expressed as (Li and Randhawa, 2005; Lukac and Plataniotis, 2004b) :

$$NCD(\mathbf{I},\hat{\mathbf{I}}) = \frac{\sum_{x=0}^{X-1}\sum_{y=0}^{Y-1}\sqrt{\sum_{k=L^*,u^*,v^*}\left(I^k_{x,y} - \hat{I}^k_{x,y}\right)^2}}{\sum_{x=0}^{X-1}\sum_{y=0}^{Y-1}\sqrt{\sum_{k=L^*,u^*,v^*}\left(I^k_{x,y}\right)^2}}\ , \tag{97}$$

   where $I^k_{x,y}$ is the level of color component k, $k \in \{L^*, u^*, v^*\}$, at the pixel having (x,y) spatial coordinates. This normalized measurement ranges from 0 (optimal demosaicing quality) to 1 (worst demosaicing quality).

Among other measurements found in the literature, let us also mention Buades et al. (2008). These authors first consider artifacts as noise which corrupts the demosaiced image, and propose an evaluation scheme based on specific characteristics of white noise. Unfortunately, the evaluation is only achieved by subjective appreciation. More interesting is the suggestion to use gray-level images for demosaicing evaluation. Indeed, color artifacts are then not only easily visually identified, but may also also be analyzed by considering the chromaticity. The rate of estimated pixels whose chromaticity is higher than a threshold reflects the propensity of a given demosaicing scheme to generate false colors.

4.4. Artifact-sensitive Measurements

The objective measurements presented above are based on an evaluation of the color estimation error. None of these measurements quantify the specific presence of each kind of artifact within the demosaiced images. Though, it would be interesting to isolate specific artifacts during the evaluation process. In this part, we present measurements which are sensitive to specific kinds of artifacts by taking their properties into account.

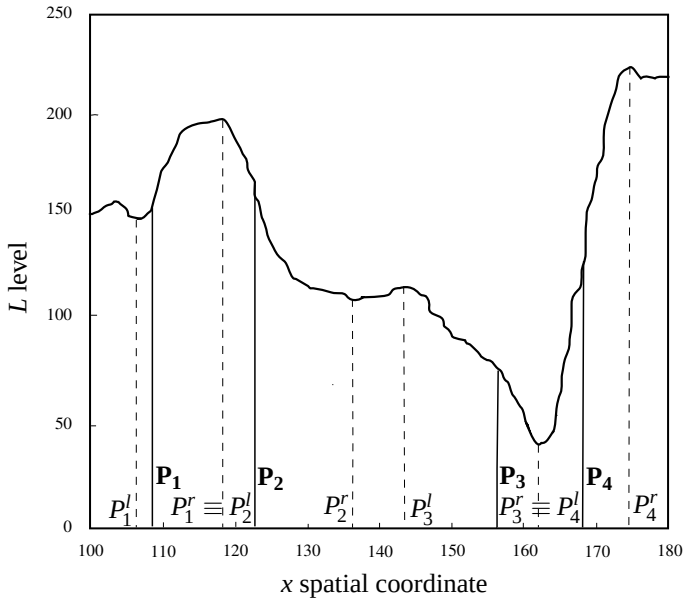


FIG. 33: Vertical edge pixels associated with their left and right pixels. Vertical edge pixels P_1, P_2, P_3 and P_4 are represented by solid lines, while pixels corresponding to extrema are located by dashed lines. The left (resp. right) extremum of a vertical edge pixel P_i is denoted P_i^l (resp. P_i^r). One single extremum may be associated with two different vertical edge pixels, for example $P_1^r \equiv P_2^l$.

4.4.1. Blurring Measurement

The blurring measurement proposed by Marziliano et al. (2004) is sensitive to the decrease of local level variations in transition areas. The authors notice that blurring corresponds to an expansion of these transition areas, and propose to measure the transition widths to quantify this artifact. The evaluation scheme analyzes the luminance planes of the original and demosaiced images, respectively denoted as L and $\hat{L}$. The transition width increase, evaluated at the same pixel locations in both images, yields an estimation of the blurring caused by demosaicing.

This blurring measurement consists in the following successive steps :

1. Apply the Sobel filter to the luminance plane L according to the horizontal direction, and threshold its output. The pixels detected in this way are called *vertical edge pixels*.
2. At each vertical edge pixel P, examine the luminance levels of pixels located on the same row as P in the luminance plane L. The pixel P^l (resp. P^r) corresponds to the first local luminance extremum located on the left (resp. the right) of P. To each vertical edge pixel P, associate in this way a pair of pixels P^l and P^r, one of them corresponding to a local luminance maximum and the other one to a minimum (see figure 33).
3. The transition width at P is defined as the difference between the x coordinates

of pixels P_l and P_r.

4. Compute the blurring measurement as the mean transition width estimated over all vertical edge pixels in the image.
5. From the spatial locations of vertical edge pixels in L – which have been detected in step 1 –, steps 2 to 4 are performed on the luminance plane $\hat{L}$ of the demosaiced image. A blurring measurement is then obtained for this plane.
6. The two measurements, obtained respectively for the original and demosaiced images, are compared in order to estimate blurring caused by the considered demosaicing scheme.

4.4.2. Zipper Effect Measurements

As far as we know, the single proposition for zipper effect measurement was given by Lu and Tan (2003). This artifact is characterized at a pixel by an increase of the minimal distance between its color and those of its neighbors. This measurement therefore relates to the original color image.

The zipper effect measurement in a demosaiced image $\hat{\mathbf{I}}$, compared with the original image $\mathbf{I}$, is computed by these successive steps :

1. At each pixel P in the original image $\mathbf{I}$, identify the neighboring pixel P' whose color is the closest to that of P in CIE $L^*a^*b^*$ color space :

$$P' = \underset{Q \in N_8}{\operatorname{argmin}} \|\mathbf{I}(P) - \mathbf{I}(Q)\|, \tag{98}$$

   where N_8 is the 8-neighborhood of P and $\|\cdot\|$ is the Euclidean distance in CIE $L^*a^*b^*$ color space. The color difference is then computed as :

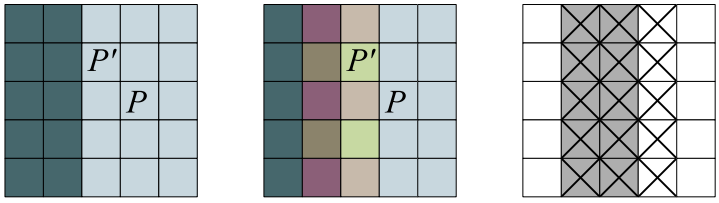
$$\Delta\mathbf{I}(P) = \left\|\mathbf{I}(P) - \mathbf{I}(P')\right\|. \tag{99}$$

2. At the same locations as P and P', compute their color difference in the demosaiced image $\hat{\mathbf{I}}$:

$$\Delta\hat{\mathbf{I}}(P) = \left\|\hat{\mathbf{I}}(P) - \hat{\mathbf{I}}(P')\right\|. \tag{100}$$

3. Compute the color difference variation $\varphi(P) = \Delta\hat{\mathbf{I}}(P) - \Delta\mathbf{I}(P)$.
4. Threshold the color difference variation, in order to detect the pixels P where zipper effect occurs. If $|\varphi(P)| > T_\varphi$, the pixel P in the demosaiced image presents a high variation of the difference between its color and that of P'. More precisely, when $\varphi(P)$ is lower than $-T_\varphi$, the demosaicing scheme has reduced the color difference between pixels P and P'. On the other hand, when $\varphi(P) > T_\varphi$, the difference between the color of P and that of P' has been highly increased in $\hat{\mathbf{I}}$ compared with $\mathbf{I}$; so, the pixel P is considered as affected by zipper effect. The authors propose to set the threshold T_φ to 2.3
5. Compute the rate of pixels affected by zipper effect in the demosaiced image :

$$ZE_{\%} = \operatorname{Card}\left\{P(x,y) \mid \varphi(P) > T_\varphi\right\}. \tag{101}$$



(a) Original image $\mathbf{I}$ (b) Demosaiced image $\hat{\mathbf{I}}$ (c) Zipper effect map

FIG. 34: Over-detection of the zipper effect by Lu and Tan's measurement (2003), in a synthetic image.
In the detection map (c), pixels affected by zipper effect are labeled as $\times$, and the ground-truth (determined by visual examination) is labeled as gray. A pixel labeled both as $\times$ and gray corresponds to a correct detection, whereas a pixel labeled only as $\times$ corresponds to an over-detection of the zipper effect.

The effectiveness of this measurement was illustrated by its authors with a synthetic image[7]. However, by applying it to images of Kodak Database, we will show in section 5.2 that it tends to over-detect zipper effect in the demosaiced images. Two reasons explain this over-detection.

First, a pixel whose color is correctly estimated and which has neighboring pixels whose colors are erroneously estimated, can be considered as being affected by zipper effect (see figure 34).

Second, we notice that all the pixels detected by Lu and Tan's measurement are not located in areas with perceptible alternating patterns which correspond to zipper effect. Indeed, all the artifacts which can increase the minimal difference between the color of a pixel and those of its neighbors do not always bear the geometric properties of zipper effect. An example of this phenomenon is found on the zipper effect detection result of figure 38c4 : almost all pixels are detected as affected by zipper effect, although the demosaiced image 38b4 does not contain this repetitive and alternating pattern.

To avoid over-detection, we propose a scheme – hereafter referred to as *directional alternation measurement* – which quantifies the level variations over three adjacent pixels along the horizontal or vertical direction in the demosaiced image.

Two reasons explain why the direction of zipper effect is mainly horizontal or vertical. Demosaicing schemes usually estimate the green color component first, then the red and blue ones by using color differences or ratios. However, along a diagonal direction in the CFA image, all the green levels are either available or missing. Since there is no alternating pattern between estimated and available levels along this diagonal direction, there are few alternating estimation errors which characterize zipper effect. Secondly, edges of objects in a natural scene tend to follow horizontal and vertical directions.

We propose to modify the selection of neighboring pixels used to decide, thanks to Lu and Tan's criterion, whether the examined pixel is affected by zipper effect. We

[7] This image is not available.

require the selected adjacent pixels to present a green alternating pattern specific to zipper effect. Moreover, this series of three adjacent pixels has to be located along transitions between homogeneous areas, so that the variations of levels associated with this transition are not taken into account.

The zipper effect detection scheme based on directional alternation, which provides a measurement for this artifact, consists in the following successive steps :

1. At a give pixel P, determine the local direction (horizontal or vertical) along which the green variations are the lowest in the original image. This direction is selected so that the green level dispersion is the lowest :

$$\sigma^x(P) = \frac{1}{3}\sum_{i=-1}^{1}\left(I^G_{x+i,y} - \mu^x(P)\right)^2 \qquad \text{and} \qquad \sigma^y(P) = \frac{1}{3}\sum_{i=-1}^{1}\left(I^G_{x,y+i} - \mu^y(P)\right)^2, \tag{102}$$

   where $\mu^x(P)$ (respectively, $\mu^y(P)$) is the mean of the green levels $I^G_{x+i,y}$ (respectively, $I^G_{x,y+i}$), $i \in \{-1,0,1\}$, in the original image $\mathbf{I}$.
   The determined direction δ is that for which the directional variance is the lowest :

$$\delta = \underset{d\in\{x,y\}}{\operatorname{argmin}}\left(\sigma^d(P)\right). \tag{104}$$

   Thanks to this step, the green levels of the three selected adjacent pixels are locally the most homogeneous.

2. Evaluate the alternation amplitude at pixel P, between the three adjacent pixels along direction δ, in the original and estimated images. When δ is horizontal, the amplitude on a plane I is computed as :

$$\alpha^x(I,P) = \left|I_{x-1,y} - I_{x,y}\right| + \left|I_{x,y} - I_{x+1,y}\right| - \left|I_{x-1,y} - I_{x+1,y}\right|, \tag{105}$$

   When δ is vertical, the amplitude is computed as :

$$\alpha^y(I,P) = \left|I_{x,y-1} - I_{x,y}\right| + \left|I_{x,y} - I_{x,y+1}\right| - \left|I_{x,y-1} - I_{x,y+1}\right|. \tag{106}$$

   When the three green levels present an alternating “high-low-high” or “low-high-low” pattern, $\alpha^\delta(I,P)$ is strictly positive, otherwise zero.

3. Compare the alternation amplitudes on the G plane of the original image $\mathbf{I}$ and that of the demosaiced image $\hat{\mathbf{I}}$. When $\alpha^\delta\left(\hat{I}^G,P\right) > \alpha^\delta\left(I^G,P\right)$, the alternation amplitude of green levels has been amplified by demosaicing along the direction δ. The pixel P is retained as a candidate pixel affected by zipper effect.

4. Apply to these candidate pixels a modified the scheme proposed by Lu and Tan, except that the neighboring pixel P' whose color is the closest to P has to be one of the two neighboring pixels along the selected direction δ.

4.4.3. *False Colors Measurement*

We also propose a measurement for the false color artifact (Yang et al., 2007). At a pixel in the demosaiced image, any mere error in the estimated value a color component can be considered as a false color. However, the human visual system cannot actually distinguish any subtle color difference which is lower than a specific threshold (Faugeras, 1979). We consider that the estimated color at a pixel is false when the absolute difference between an estimated color component and the original one is higher than a threshold T. The proposed measurement $FC_{\%}$ is the ratio between the number of pixels affected by false colors and the image size :

$$FC_{\%} = \frac{100}{XY} \text{Card} \left\{ P(x,y) \mid \max_{k=R,G,B} \left(\left| I^k_{x,y} - \hat{I}^k_{x,y} \right| \right) > T \right\} . \tag{107}$$

$FC_{\%}$ is easy to be implemented and expresses the rate of pixels affected by false colors as a measurement of the performance reached by a demosaicing scheme. Moreover, this criterion can be also used to locate pixels affected by false colors. However, as classical fidelity criteria, it requires the original image in order to compare the efficiency of demosaicing schemes.

4.5. *Measurements Dedicated to Low-level Image Analysis*

Since the demosaicing methods intend to produce "perceptually satisfying" demosaiced images, the most widely used evaluation criteria are based on the fidelity to the original images. Rather than displaying images, our long-term goal is pattern recognition by means of feature analysis. These features extracted from the demosaiced images are mostly derived from either colors or detected edges. Artifacts generated by demosaicing (mostly blurring and false colors) may affect the performance of edge detection methods applied to the demosaiced image. Indeed, blurring reduces the sharpness of edges, and false colors can give rise to irrelevant edges. That is why we propose to quantify the demosaicing performance by measuring the rates of erroneously detected edge pixels.

4.5.1. *Measurements of Sub- and Over-detected Edges.*

The edge detection procedure is sensitive to the alteration of high spatial frequencies caused by demosaicing. Indeed, low-pass filtering tends to generate blurring, and so to smooth edges. Moreover, when the demosaicing scheme generates false colors or zipper effect, it may give rise to abnormally high values of a color gradient module. The respective expected consequences are sub- and over-detection of edges. Notice that the different demosaicing algorithms are more or less efficient in avoiding to generate blurring, false color and zipper effect artifacts.

So, we propose a new evaluation scheme which performs these successive steps (Yang et al., 2007) :

1. Apply a hysteresis thresholding of the module of the color gradient proposed by Di Zenzo (1986), in order to detect edges in the original image $\mathbf{I}$. The same edge detection scheme with the same parameters is applied to the demosaiced image $\hat{\mathbf{I}}$.

   Edge detection is performed as follows :

(a) Compute the square module of the Di Zenzo gradient at each pixel in image **I** as :

$$\|\nabla \mathbf{I}\|^2 = \frac{1}{2}\left(a+c+\sqrt{(a-c)^2+4b^2}\right), \quad (108)$$

$$\theta = \frac{1}{2} arctan\left(\frac{2b}{a-c}\right), \quad (109)$$

where coefficients a, b and c are computed by approximating the partial derivatives of the image function **I** :

$$\begin{aligned} a &= \left(\frac{\partial \mathbf{I}}{\partial x}\right)^2 \approx \left(\Delta^x(I^R)\right)^2 + \left(\Delta^x(I^G)\right)^2 + \left(\Delta^x(I^B)\right)^2, \\ b &= \frac{\partial \mathbf{I}}{\partial x}\frac{\partial \mathbf{I}}{\partial y} \approx \Delta^x(I^R)\Delta^y(I^R) + \Delta^x(I^G)\Delta^y(I^G) + \Delta^x(I^B)\Delta^y(I^B), \\ c &= \left(\frac{\partial \mathbf{I}}{\partial y}\right)^2 \approx \left(\Delta^y(I^R)\right)^2 + \left(\Delta^y(I^G)\right)^2 + \left(\Delta^y(I^B)\right)^2. \end{aligned}$$

Each approximative partial derivative $\Delta^d(I^k)$, $d \in \{x,y\}$, $k \in \{R,G,B\}$, is computed thanks to the Deriche operator (Deriche, 1987).

(b) Find the local maxima of the vector gradient module $\|\nabla \mathbf{I}\|$.

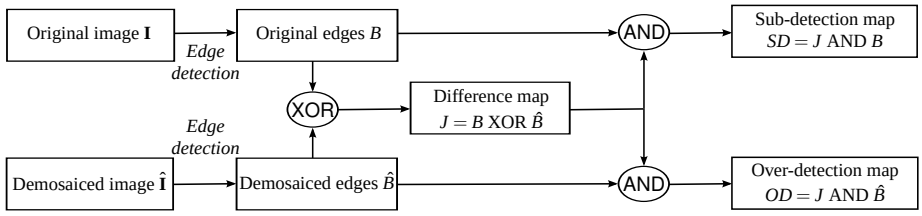
(c) Among pixels which are associated with local maxima, detect the edge pixels thanks to a hysteresis thresholding, parametrized by a low threshold T_l and a high threshold T_h.

2. Store the edge detection result for the original image in a binary edge map B, and similarly for the demosaiced image edges in $\hat{B}$. Notice that these two maps, in which edge pixels are labeled as white, may be different due to artifacts in the demosaiced image.
3. In order to quantify the influence of demosaicing on edge detection quality, we propose to follow the strategy developed by Martin et al. (2004). Edge maps B and $\hat{B}$ are compared by means of two successive operators (see figure 35a) :

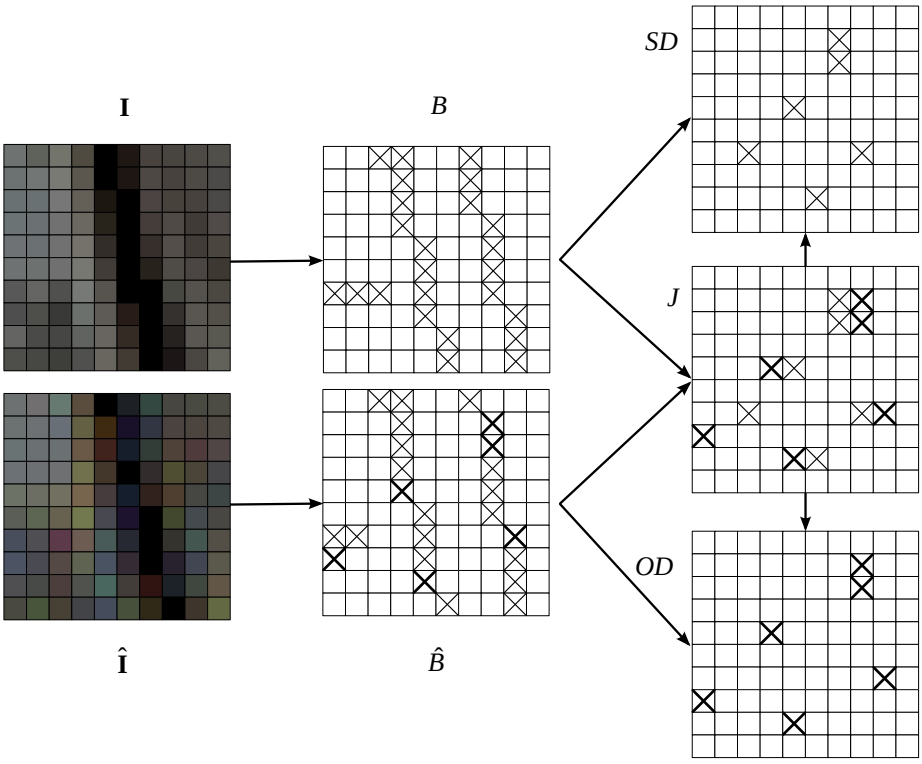
   (a) Apply the XOR logical operator to edge maps B and $\hat{B}$, in order to enhance the differences between them in a new binary map J ;

   (b) Apply the AND logical operator to maps J and B, which results in the binary sub-detected edge map SD. Similarly, the AND logical operator is applied to maps J and $\hat{B}$, which results in the binary over-detected edge map OD.

   Pixels labeled as white in map SD are edge pixels which are detected in the original image **I** but undetected in the demosaiced image $\hat{\mathbf{I}}$. Pixels labeled as white in the image OD are edge pixels erroneously detected in the demosaiced image $\hat{\mathbf{I}}$, compared with edge pixels detected in **I**.



(a) General scheme



(b) Example

FIG. 35: Steps to measure the quality of edge detection.
In subfigure (b), over-detected edge pixels are labeled as **×** (in bold typeface) in $\hat{B}$, J and OD, in order to distinguish them from sub-detected edge pixels (labeled as $\times$).

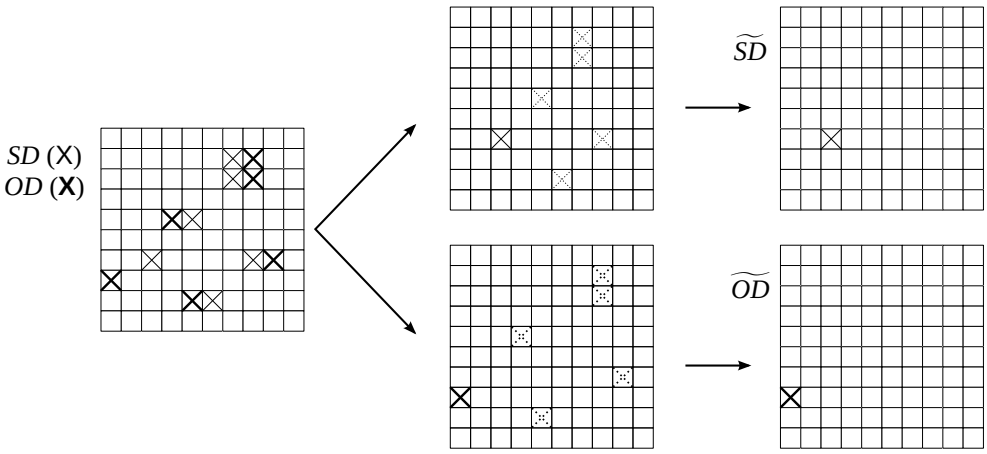


FIG. 36: Computing $\widetilde{SD}$ and $\widetilde{OD}$ from SD and OD, on an example. Pixels labeled as dotted $\times$ belong to pairs of shifted edge pixels, and are dropped out in the final detection maps.

4. Compute the rates of sub- and over-detected edge pixels respectively as :

$$SD_{\%} = \frac{100}{XY}\text{Card}\left\{P(x,y) \mid SD_{x,y} \neq 0\right\}, \quad (110)$$

$$OD_{\%} = \frac{100}{XY}\text{Card}\left\{P(x,y) \mid OD_{x,y} \neq 0\right\}. \quad (111)$$

Finally, the rate of erroneously detected edge pixels is expressed as $ED_{\%} = SD_{\%} + OD_{\%}$.

4.5.2. Measurements Based on Shifted Edges

By visually examining the map J in figure 35b, we notice the presence of many pairs of adjacent edge pixels. In such edge pairs, one pixel is detected in B only (i.e. sub-detected), and the other one in $\hat{B}$ only (i.e. over-detected). For example, the map J of figure 35b presents five pairs of adjacent pixels composed of a sub-detected edge pixel (labeled as $\times$) and an over-detected edge pixel (labeled as $\times$ in bold typeface). These cases do not result from a bad edge detection, but from a spatial shift of edge pixels between the original and demosaiced images. A sub-detected (respectively, over-detected) edge pixel is shifted when at least one of its neighbors is an over-detected (respectively, sub-detected) edge pixel. Such pairs of pixels are hereafter called *pairs of shifted (edge) pixels*.

In order to characterize the effect of demosaicing on edge detection precisely, we want to distinguish pairs of shifted edge pixels from other edge pixels. For this purpose, we represent unshifted sub- and over-detected edge pixels as two binary maps respectively denoted as $\widetilde{SD}$ and $\widetilde{OD}$, and defined as :

$$\widetilde{SD}_{x,y} \neq 0 \quad \Leftrightarrow \quad SD_{x,y} \neq 0 \wedge \left(\nexists Q(x',y') \in N_8\left(P(x,y)\right) \middle| OD_{x',y'} \neq 0 \right), \quad (112)$$

$$\widetilde{OD}_{x,y} \neq 0 \quad \Leftrightarrow \quad OD_{x,y} \neq 0 \wedge \left(\nexists Q(x',y') \in N_8\left(P(x,y)\right) \middle| SD_{x',y'} \neq 0 \right), \quad (113)$$

where symbol $\wedge$ represents the logical AND operator.

Figure 36 illustrates, from the example of figure 35, how maps $\widetilde{SD}$ and $\widetilde{OD}$ are obtained. In this figure, maps SD and OD used to build $\widetilde{SD}$ and $\widetilde{OD}$ are superimposed in order to highlight the pairs of shifted edge pixels.

From the two binary maps $\widetilde{SD}$ and $\widetilde{OD}$, we compute the rates of sub- and over-detected unshifted edge pixels as :

$$\widetilde{SD}_{\%} = \frac{100}{XY} \mathrm{Card}\left\{ P(x,y) \mid \widetilde{SD}_{x,y} \neq 0 \right\}, \quad (114)$$

$$\widetilde{OD}_{\%} = \frac{100}{XY} \mathrm{Card}\left\{ P(x,y) \mid \widetilde{OD}_{x,y} \neq 0 \right\}. \quad (115)$$

These rates are used to evaluate precisely the quality of edge detection in demosaiced images.

4.6. Conclusion

In this section, we have presented the techniques of objective evaluation of demosaicing quality. For this purpose, we have first presented the most occurred artifacts caused by demosaicing. Blurring, false colors and zipper effect damage the quality of the demosaiced images.

Then, we have presented classical criteria which total the errors between the original and estimated colors through the image. These criteria have some limits since they provide a global estimation of the demosaicing quality and do not reflect the judgment of an observer. Indeed, they do not quantify the occurrences of artifacts which can be identified by an observer. Therefore, we have described measurements dedicated to three kinds of artifacts.

In the computer vision context, most images are acquired by color mono-sensor cameras in order to be automatically processed. So, the quality of demosaicing affects the quality of low-level image analysis schemes. That is the reason why we have proposed criteria which are based on the quality of edge detection.

5. Quality Evaluation Results

5.1. Results of Classical Criteria

The quality of demosaicing results, achieved by the ten methods detailed in section 2, has been first evaluated thanks to classical criteria. For this purpose, the twelve mostly used images of Kodak benchmark database are considered (see figure 37)[8]. These images, all of size 768×512 pixels, have been selected in order to present a significant variety of homogeneous regions, colors and textured areas.

Table 3 displays the results obtained with criteria which measure the fidelity of each demosaiced image to its corresponding original image, namely the mean absolute error (*MAE*, expression (82)), the peak signal-to-noise ratio (*PSNR*, expression (85)) and the correlation criterion (*C*, expression (87)). Table 4 shows, for the same images and demosaicing schemes, the results obtained with perceptual criteria, namely the estimation error in CIE $L^*a^*b^*$ color space ($\Delta E^{L^*a^*b^*}$, expression (92)), the estimation error in S-CIE $L^*a^*b^*$ color space ($\Delta E^{S\text{-}L^*a^*b^*}$) and the criterion of normalized color difference (*NCD*, expression (97)) between the demosaiced image and its original image.

These two tables show that for a given method, the performances measured with a specific criterion vary from an image to another one. This confirms that obtaining a good color estimation from the CFA image is all the more difficult as the image is rich in high spatial frequency areas. For instance, the *PSNR* of images demosaiced by bilinear interpolation ranges from 24.5 dB for image 4 (“House”), which contains a lot of high frequency areas, to 36 dB for image 1 (“Parrots”), which contains a lot of homogeneous regions.

It can be noticed that the two methods which chiefly use the frequency domain provide better results than those which only scan the spatial domain. Moreover, the method proposed by Dubois (2005) achieves the best average results over the twelve images, whatever the considered criterion. We also notice that the different criteria provide similar performance rankings for the methods on a given image.

5.2. Results of Artifact-sensitive Measurements

5.2.1. Zipper Effect Measurements

In order to compare the relevance of the results provided by the two zipper effect measurements described in section 4.4.2, we propose to use the following procedure. First, a ground truth is built for the zipper effect by visually examining the demosaiced image and defining whether each pixel is affected by zipper effect or not. Then, the two measurements are applied, in order to provide binary maps where pixels which are affected by zipper effect are labeled as white. A final comparison step of these binary maps with the ground truth quantifies the performance of each objective measurement, by counting pixels where zipper effect is correctly detected, sub-detected and over-detected.

Figure 38 displays the results on four image extracts of size 10×10 pixels. It shows that the directional alternation measurement generally fits better with the ground truth

[8] This database is available at `http://www.math.purdue.edu/~lucier/PHOTO_CD`

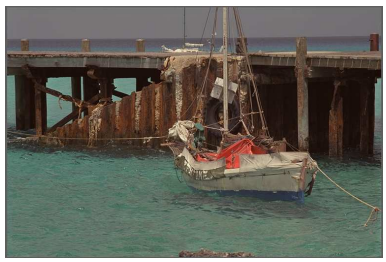
1. Parrots

2. Sailboat

3. Windows

4. Houses

5. Race



6. Pier

7. Island

8. Lighthouse

9. Plane

10. Cape

11. Barn

12. Chalet

FIG. 37: The twelve benchmark images picked up from Kodak database. Images 5 and 8 are presented vertically for illustration purpose, but have been analyzed in landscape orientation.

Image	Criterion	Bilinear	Cst. Hue	Hamilton	Wu	Cok	Kimmel	Li	Gunturk	Dubois	Lian
1	*MAE*	1.542	1.358	0.938	0.949	1.257	1.784	1.379	0.877	0.879	**0.796**
	PSNR	36.256	38.082	42.868	42.984	39.069	31.883	38.132	43.186	43.259	**44.199**
	C	0.9966	0.9978	0.9993	0.9993	0.9982	0.9912	0.9978	0.9993	0.9993	**0.9995**
2	*MAE*	4.352	3.381	1.829	1.565	2.897	2.241	2.515	1.339	**1.154**	1.415
	PSNR	28.956	31.396	36.324	37.831	32.561	34.418	33.499	39.951	**41.433**	39.303
	C	0.9830	0.9905	0.9970	0.9978	0.9928	0.9952	0.9942	0.9987	**0.9990**	0.9984
3	*MAE*	1.978	1.578	0.980	0.994	1.407	1.264	1.484	0.907	0.900	**0.786**
	PSNR	34.454	36.779	41.773	41.641	37.915	38.620	37.111	42.713	43.062	**43.832**
	C	0.9909	0.9946	0.9983	0.9982	0.9958	0.9965	0.9950	0.9987	0.9987	**0.9989**
4	*MAE*	7.329	5.655	2.629	2.607	3.986	3.077	4.130	2.055	2.022	**1.975**
	PSNR	24.551	27.350	33.409	33.535	29.885	31.858	29.588	36.452	**36.479**	36.445
	C	0.9596	0.9799	0.9950	0.9951	0.9888	0.9928	0.9881	0.9975	**0.9975**	0.9975
5	*MAE*	2.276	1.822	1.112	1.078	1.591	1.230	1.556	0.896	0.895	**0.860**
	PSNR	33.611	36.120	41.430	41.795	37.701	39.659	37.515	43.237	43.354	**43.785**
	C	0.9863	0.9926	0.9978	0.9980	0.9949	0.9967	0.9946	0.9985	0.9986	**0.9987**
6	*MAE*	3.589	2.857	1.605	1.511	2.404	1.949	2.370	1.247	**1.167**	1.215
	PSNR	30.191	32.400	37.353	37.748	33.579	35.344	33.372	40.409	**40.894**	40.399
	C	0.9783	0.9874	0.9960	0.9963	0.9905	0.9935	0.9900	0.9980	**0.9982**	0.9980
7	*MAE*	2.880	2.264	1.263	1.084	1.931	1.518	1.652	0.964	**0.826**	1.022
	PSNR	32.341	34.719	39.713	41.613	36.141	37.788	37.451	42.913	**44.680**	42.144
	C	0.9861	0.9921	0.9975	0.9984	0.9944	0.9961	0.9958	0.9988	**0.9992**	0.9986
8	*MAE*	3.849	3.079	1.571	1.546	2.344	1.874	2.284	1.234	**1.164**	1.195
	PSNR	29.186	31.716	38.419	38.594	34.663	36.172	34.708	**42.913**	41.547	41.072
	C	0.9775	0.9875	0.9973	0.9974	0.9936	0.9956	0.9938	0.9985	**0.9987**	0.9986
9	*MAE*	2.362	1.929	1.306	1.318	1.769	1.394	1.802	1.043	1.114	**0.994**
	PSNR	32.565	34.931	39.462	39.347	35.985	38.181	35.601	42.030	41.735	**42.353**
	C	0.9973	0.9984	0.9995	0.9994	0.9988	0.9993	0.9987	0.9997	0.9997	**0.9997**
10	*MAE*	3.772	2.936	1.840	1.801	2.661	1.969	2.739	1.311	**1.290**	1.319
	PSNR	29.557	31.960	36.542	36.643	32.891	35.202	32.549	**40.220**	40.172	39.972
	C	0.9769	0.9870	0.9955	0.9955	0.9895	0.9939	0.9887	**0.9981**	0.9981	0.9980
11	*MAE*	3.164	2.497	1.701	1.741	2.346	1.971	2.535	1.442	1.368	**1.326**
	PSNR	31.433	33.718	37.746	37.455	34.560	35.995	33.802	39.217	39.575	**39.963**
	C	0.9849	0.9909	0.9964	0.9962	0.9925	0.9949	0.9913	0.9975	0.9977	**0.9979**
12	*MAE*	4.366	3.317	2.057	1.965	3.091	2.244	3.310	1.530	**1.453**	1.469
	PSNR	27.564	29.938	33.381	34.237	29.957	32.196	29.333	36.630	**37.690**	36.687
	C	0.9752	0.9859	0.9936	0.9948	0.9859	0.9915	0.9838	0.9970	**0.9976**	0.9970
Avg.	*MAE*	3.455	2.723	1.569	1.513	2.307	1.876	2.313	1.237	**1.186**	1.198
	PSNR	30.889	33.259	38.202	38.619	34.575	35.610	34.388	40.823	**41.157**	40.846
	C	0.9827	0.9904	0.9969	0.9972	0.9930	0.9947	0.9926	0.9983	**0.9985**	0.9984

TAB. 3: Demosaicing quality results, for twelve color images from Kodak database, according to fidelity criteria : mean absolute error (*MAE*), peak signal-to-noise ratio (*PSNR*, in decibels), and correlation (*C*) between the original image and the demosaiced image. For each image and each criterion, the best result is written in bold typeface. The tested methods are : 1. Bilinear interpolation – 2. Constant-hue-based interpolation (Cok, 1987) – 3. Gradient-based method (Hamilton and Adams, 1997) – 4. Component-consistent scheme (Wu and Zhang, 2004) – 5. Method based on template matching (Cok, 1986) – 6. Adaptive weighted-edge method (Kimmel, 1999) – 7. Covariance-based method (Li and Orchard, 2001) – 8. Alternating projection method (Gunturk et al., 2002) – 9. Frequency selection method (Dubois, 2005) – 10. Method based on frequency and spatial analyses (Lian et al., 2007).

Image	Criterion	Bilinear	Cst. Hue	Hamilton	Wu	Cok	Kimmel	Li	Gunturk	Dubois	Lian
1	$\Delta E^{L^*a^*b^*}$	1.439	1.289	1.002	1.010	1.229	1.655	1.387	0.969	0.952	**0.899**
	$\Delta E^{S\text{-}L^*a^*b^*}$	2.605	2.605	2.318	2.268	2.682	5.701	3.537	2.193	**1.967**	2.007
	NCD	0.0098	0.0089	0.0067	0.0068	0.0083	0.0119	0.0094	0.0064	0.0064	**0.0060**
2	$\Delta E^{L^*a^*b^*}$	3.382	2.562	1.538	1.335	2.275	1.772	2.078	1.196	**1.078**	1.223
	$\Delta E^{S\text{-}L^*a^*b^*}$	6.477	5.965	3.756	2.954	5.360	4.653	4.578	3.079	**2.440**	3.021
	NCD	0.0251	0.0194	0.0113	0.0099	0.0169	0.0136	0.0152	0.0089	**0.0079**	0.0091
3	$\Delta E^{L^*a^*b^*}$	2.048	1.663	1.132	1.148	1.492	1.491	1.653	1.108	1.066	**0.981**
	$\Delta E^{S\text{-}L^*a^*b^*}$	3.715	3.483	2.659	2.615	3.339	4.283	3.990	2.594	2.280	**2.229**
	NCD	0.0140	0.0114	0.0077	0.0078	0.0101	0.0102	0.0112	0.0074	0.0072	**0.0066**
4	$\Delta E^{L^*a^*b^*}$	5.467	4.246	2.167	2.099	3.138	2.356	3.315	1.735	1.676	**1.652**
	$\Delta E^{S\text{-}L^*a^*b^*}$	11.293	10.635	5.729	5.166	7.918	6.125	7.886	4.850	4.507	**4.327**
	NCD	0.0441	0.0338	0.0172	0.0169	0.0249	0.0193	0.0261	0.0140	0.0136	**0.0132**
5	$\Delta E^{L^*a^*b^*}$	1.780	1.474	0.965	0.931	1.273	1.040	1.299	0.861	0.843	**0.816**
	$\Delta E^{S\text{-}L^*a^*b^*}$	3.925	3.824	2.753	2.661	3.401	3.462	3.344	2.437	2.361	**2.304**
	NCD	0.0139	0.0114	0.0074	0.0072	0.0099	0.0082	0.0100	0.0065	0.0064	**0.0062**
6	$\Delta E^{L^*a^*b^*}$	3.511	2.762	1.729	1.641	2.419	1.943	2.485	1.393	**1.334**	1.343
	$\Delta E^{S\text{-}L^*a^*b^*}$	6.883	6.417	4.333	3.809	5.781	4.806	5.675	3.589	**3.209**	3.323
	NCD	0.0261	0.0209	0.0128	0.0122	0.0179	0.0151	0.0183	0.0104	**0.0099**	0.0100
7	$\Delta E^{L^*a^*b^*}$	2.671	2.047	1.259	1.088	1.789	1.407	1.592	1.021	**0.895**	1.051
	$\Delta E^{S\text{-}L^*a^*b^*}$	5.231	4.808	3.135	2.496	4.254	3.563	3.580	2.597	**1.991**	2.635
	NCD	0.0206	0.0161	0.0096	0.0083	0.0138	0.0113	0.0121	0.0079	**0.0068**	0.0081
8	$\Delta E^{L^*a^*b^*}$	3.338	2.629	1.561	1.526	2.170	1.806	2.195	1.260	**1.188**	1.224
	$\Delta E^{S\text{-}L^*a^*b^*}$	6.474	5.984	3.811	3.465	5.039	4.404	4.857	3.208	**2.860**	2.963
	NCD	0.0243	0.0193	0.0111	0.0110	0.0156	0.0133	0.0157	0.0090	**0.0085**	0.0087
9	$\Delta E^{L^*a^*b^*}$	2.155	1.725	1.221	1.208	1.613	1.277	1.709	0.996	1.005	**0.959**
	$\Delta E^{S\text{-}L^*a^*b^*}$	4.568	4.136	3.175	2.984	3.909	3.346	4.663	2.791	2.697	**2.478**
	NCD	0.0150	0.0122	0.0086	0.0086	0.0113	0.0093	0.0119	0.0071	0.0072	**0.0068**
10	$\Delta E^{L^*a^*b^*}$	3.259	2.524	1.705	1.652	2.356	1.696	2.517	1.273	**1.261**	1.278
	$\Delta E^{S\text{-}L^*a^*b^*}$	6.239	5.839	4.234	3.826	5.555	4.060	5.694	3.321	**3.107**	3.140
	NCD	0.0251	0.0197	0.0131	0.0128	0.0182	0.0137	0.0192	0.0099	**0.0097**	0.0098
11	$\Delta E^{L^*a^*b^*}$	2.724	2.152	1.584	1.602	2.065	1.822	2.284	1.416	1.319	**1.303**
	$\Delta E^{S\text{-}L^*a^*b^*}$	5.175	4.747	3.898	3.738	4.852	4.690	5.371	3.631	**3.157**	3.191
	NCD	0.0195	0.0157	0.0114	0.0116	0.0149	0.0133	0.0165	0.0101	0.0095	**0.0093**
12	$\Delta E^{L^*a^*b^*}$	3.402	2.620	1.736	1.655	2.482	1.814	2.730	1.380	1.318	**1.317**
	$\Delta E^{S\text{-}L^*a^*b^*}$	6.286	5.870	4.341	3.920	5.965	4.384	6.371	3.564	**3.135**	3.193
	NCD	0.0258	0.0200	0.0132	0.0127	0.0188	0.0142	0.0206	0.0105	0.0101	**0.0100**
Avg.	$\Delta E^{L^*a^*b^*}$	2.931	2.308	1.467	1.408	2.025	1.673	2.104	1.217	**1.161**	1.170
	$\Delta E^{S\text{-}L^*a^*b^*}$	5.739	5.359	3.678	3.325	4.838	4.456	4.962	3.154	**2.809**	2.901
	NCD	0.0219	0.0174	0.0108	0.0105	0.0150	0.0128	0.0155	0.0090	**0.0086**	0.0086

TAB. 4: Demosaicing quality results, for twelve color images from Kodak database, according to perceptual criteria : estimation error in CIE $L^*a^*b^*$ color space ($\Delta E^{L^*a^*b^*}$), estimation error in S-CIE $L^*a^*b^*$ color space ($\Delta E^{S\text{-}L^*a^*b^*}$), and criterion of normalized color difference (*NCD*). For each image and each criterion, the best result (i.e. lowest value) is written in bold typeface. Images and tested methods are the same as in table 3. The illuminant used for (X,Y,Z) transform is the standard CIE D65, which corresponds to daylight.

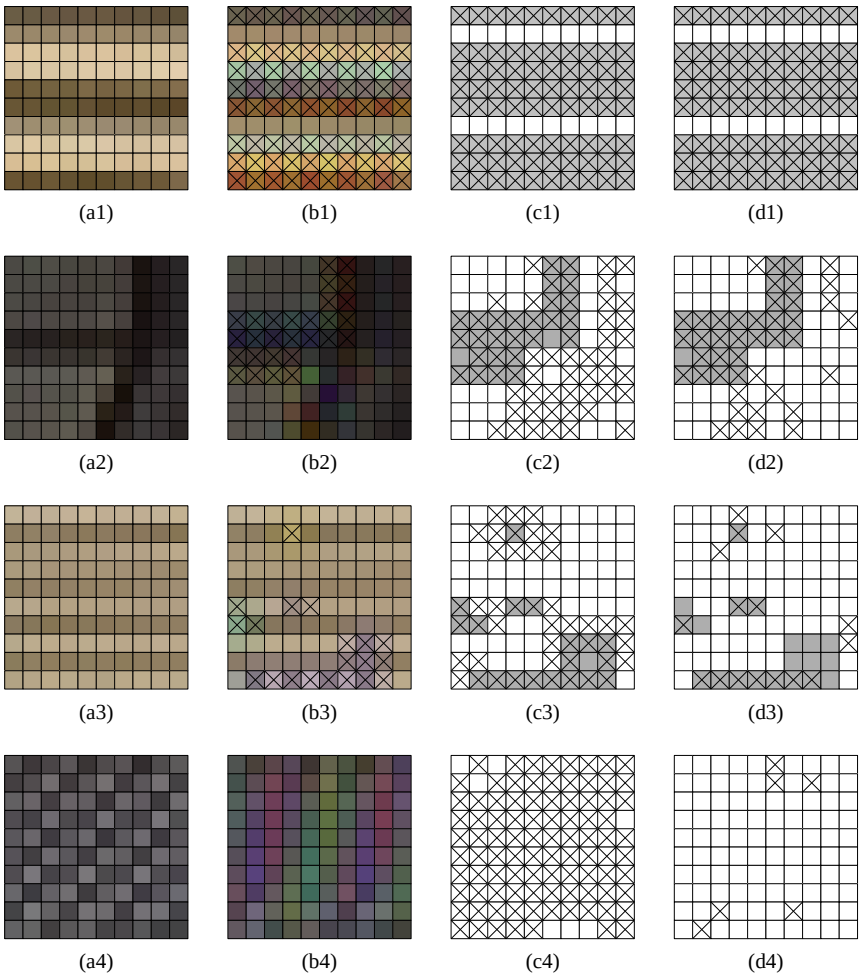


FIG. 38: Zipper effect detection in four Kodak image extracts, according to two measurements.
(a1)–(a4) : original extracts. (b1)–(b4) : demosaiced extracts. Last two columns : pixels affected by zipper effect, according to Lu and Tan's criterion (c1)–(c4) and to the directional alternation (d1)–(d4).
Pixels affected by zipper effect are labeled as $\times$. They correspond to ground truth in images (b1)–(b4). In images (c1)–(d4), the ground truth is reproduced as gray-labeled pixels. So, pixels where the zipper effect is well detected are both labeled as $\times$ and gray. Pixels where the zipper effect is sub-detected (respectively, over-detected) are labeled only as $\times$ (respectively, only as gray). Images (b1) and (b2) are estimated by bilinear interpolation, (b3) and (b4) by Hamilton and Adams' (1997) gradient-based method.

Image	Well-detected		Sub-detected		Over-detected	
	Lu and Tan	Directional alternation	Lu and Tan	Directional alternation	Lu and Tan	Directional alternation
(a1)	100	100	0	0	0	0
(a2)	58	83	2	1	40	16
(a3)	72	86	1	9	27	5
(a4)	7	94	0	0	93	6
Total	237	363	3	10	160	27

TAB. 5: Comparison between the measurements quantifying zipper effect, proposed by Lu and Tan (2003) and based on the directional alternation. Values correspond to the numbers of well-detected, sub-detected and over-detected pixels affected by this artifact in the four image extracts of figure 38.

than Lu and Tan's measurement does. This remark is confirmed numerically by comparing the numbers of well-detected, sub-detected and over-detected pixels affected by zipper effect in the four images. The results in table 5 show that the measurement based on directional alternation generally provides higher well-detected pixel rates than the one proposed by Lu and Tan. Indeed, the latter over-detects zipper effect whereas the measurement based on directional alternation tends to slightly sub-detect this artifact.

Finally, we have compared the demosaicing schemes according to the measurement based on directional alternation. Table 6 shows that the results are similar to those obtained with classical criteria, presented in tables 3 and 4 : bilinear interpolation always generates the highest amount of zipper effect, whereas the scheme proposed by Lian et al. (2007) is overall the most efficient. However, by examining table 6 in detail, we notice that in images with few high spatial frequencies (number 2-"Sailboat" and 7-"Island"), the method proposed by Dubois tends to generate less zipper artifact than Lian et al.'s method does. Generally speaking, these results show that the methods which analyze the frequency domain generate less zipper effect than those which scan the image plane (Menon et al., 2006).

5.2.2. False colors

As described in section 4.4.3, the estimated color at a pixel is taken as false when the absolute difference between an estimated color component and the original one is higher than a threshold T (see equation (107)). Since adjusting this threshold is not easy, we compare the performance reached by a set of ten demosaicing schemes applied to twelve images of the Kodak database, when T varies from 10 to 25 with an incremental step of 5. Figure 39 shows both the evolution of the average rate of false colors with respect to T for a given scheme, and the rates of false colors generated by the considered schemes for a given value of T. As expected, the rate of false colors decreases when T increases. More interestingly, the relative ranking of demosaicing methods with respect to the number of false colors is consistent with both rankings provided by classical fidelity criteria and by measurements based on zipper effect.

Image	Bilinear	Cst. Hue	Hamilton	Wu	Cok	Kimmel	Li	Gunturk	Dubois	Lian
1	4.317	1.939	0.623	0.822	0.735	4.408	3.068	0.893	0.861	**0.345**
2	22.567	12.761	2.656	2.082	4.903	2.464	7.157	0.682	**0.487**	0.590
3	8.793	4.581	1.257	1.626	1.374	1.795	4.093	1.664	1.278	**0.546**
4	35.932	25.164	4.485	5.393	7.214	5.023	14.031	2.402	2.351	**1.610**
5	9.023	4.226	0.610	0.581	1.110	0.658	2.069	0.664	0.482	**0.192**
6	19.876	10.707	2.955	3.405	3.986	2.797	7.868	1.562	1.441	**0.826**
7	18.483	10.124	1.954	1.213	3.730	1.990	4.579	0.391	**0.177**	0.436
8	18.216	11.672	2.369	3.051	3.811	2.122	7.213	0.850	0.727	**0.617**
9	9.459	5.618	1.695	2.192	2.367	1.537	5.335	0.714	0.709	**0.422**
10	15.425	9.976	3.021	3.473	4.003	2.475	8.548	0.984	0.967	**0.685**
11	12.816	6.331	1.809	2.726	2.840	1.835	7.083	1.166	0.962	**0.510**
12	18.729	10.107	2.735	3.461	3.761	2.269	9.256	1.285	1.076	**0.803**
Avg.	16.136	9.434	2.181	2.502	3.319	2.448	6.692	1.105	0.960	**0.632**

TAB. 6: Rates $ZE_{\%}$ of pixels affected by zipper effect, according to the measurement based on directional alternation. The images and tested methods are the same as in table 3.

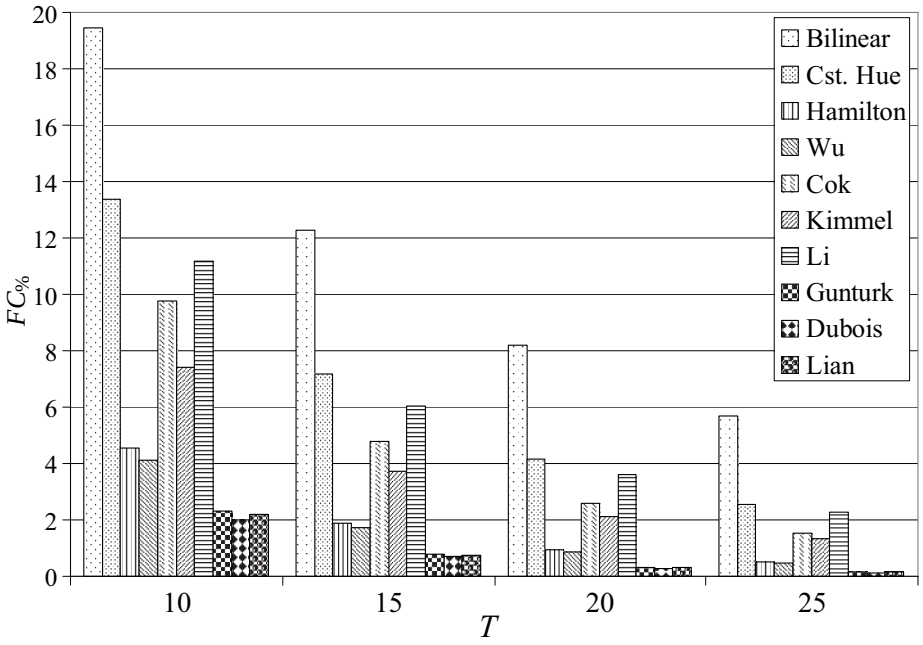


FIG. 39: Average rates of false colors $FC_{\%}$ with respect to the detection threshold T. The twelve considered images and ten tested methods are the same as in table 3.

5.3. *Discussion*

The most widely used criteria for the evaluation of demosaicing quality are *MSE* and *PSNR*, the latter being a logarithmic form of the *MSE* criterion. Several reasons explain why most of the authors use these criteria (Wang and Bovik, 2006). First, these functions are easy to be implemented and their derivatives can be estimated. They may therefore be integrated into an optimization scheme. Second, the *PSNR* criterion has a real physical meaning – namely, the maximal energy of the signal with respect to errors generated by demosaicing –, which can also be analyzed in the frequency domain.

However, the *PSNR* criterion provides a general estimation of the demosaicing quality, but does not really reflect the human judgment. For example, an observer would prefer an image containing a large number of pixels with estimated colors close to the original ones, than an image containing a reduced number of pixels affected by visible artifacts. But *MSE* and *PSNR* criteria could provide identical values in both cases, since they do not discriminate the characteristics of different artifacts in the demosaiced image. These objective measurements have been criticized (Wang and Bovik, 2009) since they cannot evaluate the image alteration as a human observer does (Eskicioglu and Fisher, 1995).

The alternative criteria ΔE of estimation errors in the CIE $L^*a^*b^*$ and S-CIE $L^*a^*b^*$ color spaces are the most widely used perceptual criteria (Zhang and Wandell, 1997). They are based on perceptually uniform color spaces as an attempt to represent the human perception, but require prior knowledge about the illuminant and the reference white used during image acquisition. Since the acquisition conditions are not always known, the quality of these measurements may be biased.

5.4. *Experimental Results for Edge Detection*

The demosaicing performance has been evaluated with respect to the quality of edge detection thanks to measurements detailed in section 4.5. Table 7 displays the average rates of sub-detected ($SD_{\%}$), over-detected ($OD_{\%}$) and erreneously detected ($ED_{\%} = SD_{\%} + OD_{\%}$) edge pixels. These values have been computed over the twelve Kodak images previously considered, and for the ten classical demosaicing schemes. Moreover, this table displays the average rates $\widetilde{SD}_{\%}$, $\widetilde{OD}_{\%}$ and $\widetilde{ED}_{\%}$ which take into account only unshifted edge pixels. The lowest values correspond to the best demosaicing quality according to these edge-dedicated measurements.

By examining the average rates $ED_{\%}$ and $\widetilde{ED}_{\%}$, similar conclusions can be drawn about the performances of demosaicing schemes. The methods which privilege the frequency domain allow to obtain better edge detection quality than the other methods do. Besides, the methods proposed by Dubois and by Lian et al. provide the lowest error rates in both edge and unshifted edge detection. These demosaicing schemes are therefore the most apt to be coupled with edge detection procedures based on color gradient.

Moreover, we notice that the ranking of the ten tested demosaicing schemes with respect to $OD_{\%}$ and $SD_{\%}$ is relatively consistent with the ranking obtained with measurements $\widetilde{OD}_{\%}$ and $\widetilde{SD}_{\%}$. However, the rate of over-detected unshifted pixels is the lowest for bilinear interpolation. This suprising performance result can be explained

Meas.	Bilinear	Cst. Hue	Hamilton	Wu	Cok	Kimmel	Li	Gunturk	Dubois	Lian
$SD_{\%}$	3.673	2.090	1.528	1.561	1.882	1.983	2.265	1.422	**1.278**	1.323
$OD_{\%}$	2.257	1.945	1.504	1.522	1.818	1.802	2.319	1.242	**1.199**	1.263
$ED_{\%}$	5.930	4.035	3.032	3.083	3.700	3.785	4.584	2.664	**2.477**	2.586
$\widetilde{SD}_{\%}$	1.945	1.109	0.881	0.877	1.032	1.077	1.094	0.888	**0.774**	0.803
$\widetilde{OD}_{\%}$	**0.663**	0.979	0.855	0.842	0.974	0.912	1.156	0.713	0.697	0.748
$\widetilde{ED}_{\%}$	2.608	2.088	1.736	1.719	2.006	1.989	2.250	1.601	**1.471**	1.551

TAB. 7: Average rates of sub-detected edge pixels ($SD_{\%}$), of over-detected edge pixels ($OD_{\%}$) and erreneously detected pixels ($ED_{\%} = SD_{\%} + OD_{\%}$). Average rates of sub-detected unshifted edge pixels ($\widetilde{SD}_{\%}$), of over-detected unshifted edge pixels ($\widetilde{OD}_{\%}$) and of unshifted edge pixels that are erreneously detected ($\widetilde{ED}_{\%} = \widetilde{SD}_{\%} + \widetilde{OD}_{\%}$). The low and high thresholds used for hysteresis thresholding are set to 1 and 6, respectively. The twelve considered images and ten tested methods are the same as in table 3.

by both strong blurring and zipper effect generated by this demosaicing method. Indeed, blurring induces fewer detected edge pixels, and zipper effect mainly induces pairs of shifted edge pixels.

For each of the other methods, the rates of sub- and over-detected edge pixels are overall similar. Moreover, their ranking is almost the same as the one obtained with the previous criteria.

In table 7, we also notice that more than the half of sub- and over-detected edge pixels according to measurements $SD_{\%}$ and $OD_{\%}$ are not retrieved with measurements $\widetilde{SD}_{\%}$ and $\widetilde{OD}_{\%}$. That means that shifted edges strongly contribute to the dissimilarity between edges detected in the original and demosaiced images.

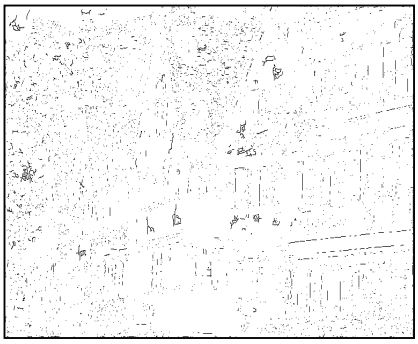
Edge pixels are sub-detected because the color gradient module used to detect edges decreases with blurring in demosaiced images. The over-detected edge pixels correspond to an increase of the color gradient module in case of zipper effect or false colors. These new rates of sub- and over-detected pixels $\widetilde{SD}_{\%}$ and $\widetilde{OD}_{\%}$ are able to reflect the artifacts caused by demosaicing. From table 7, we can evaluate the influence, on edge detection, of the demosaicing strategies implemented in the tested methods. Both methods using bilinear interpolation and hue constancy estimate the pixel colors without exploiting spatial correlation. Hence, they generate more artifacts than the three other methods which exploit spatial correlation, and provide higher rates of sub- and over-detected edge pixels.

All in all, sub- and over-detected edge pixels often coincide with artifacts. Figure 40 shows images which are demosaiced by two different schemes, and the respective maps of sub- and over-detected unshifted edge pixels ($\widetilde{SD}$ and $\widetilde{OD}$). We notice that demosaicing influences the edge detection more significantly in areas with high spatial frequencies and that the artifacts are also mainly located in these areas.

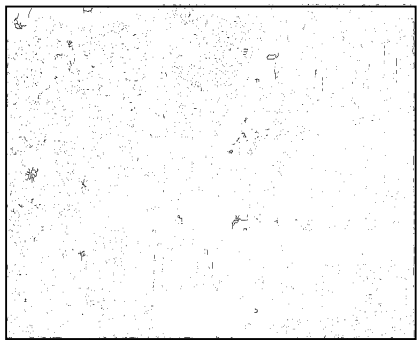
Zipper effect often decreases the variation of levels in transition areas between homogeneous regions. Hence, zipper effect tends to decrease the gradient module, so that the norm of local maxima becomes lower than the high threshold T_h used by the

(a) Image demosaiced by bilinear interpolation

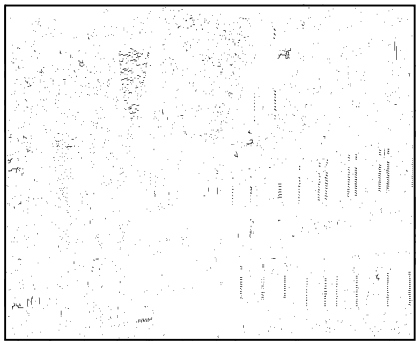
(b) Image demosaiced by Hamilton and Adams



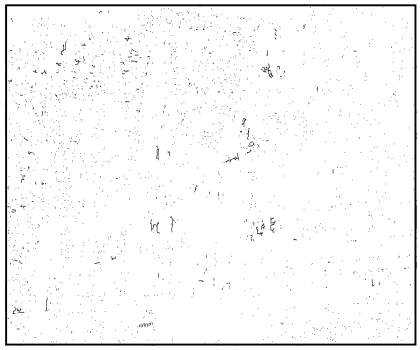
(c) Sub-detected edge pixels $\widetilde{SD}$ in image (a)



(d) Sub-detected edge pixels $\widetilde{SD}$ in image (b)

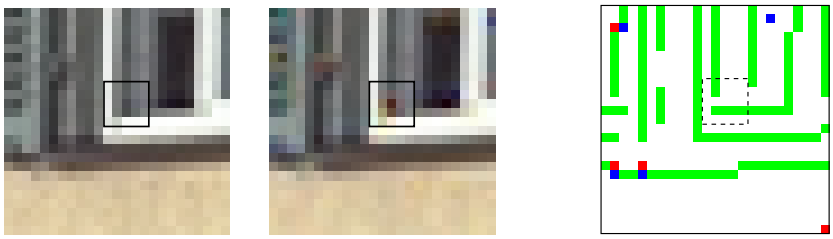


(e) Over-detected edge pixels $\widetilde{OD}$ in image (a)



(f) Over-detected edge pixels $\widetilde{OD}$ in image (b)

FIG. 40: Sub- and over-detected unshifted edge pixels, for two demosaicing schemes : bilinear interpolation and the gradient-based method proposed by Hamilton and Adams (1997).



(a) Original image $\mathbf{I}$ (b) Demosaiced image $\hat{\mathbf{I}}$ (c) Comparison between detected edge pixels (green : coinciding ; blue : sub-detected ; red : over-detected)

FIG. 41: Example of edge pixels which are not modified by pixels affected by false colors, on an image demosaiced by the scheme proposed by Hamilton and Adams (1997).

hysteresis thresholding (see page 65). This explains why zipper effect causes edge sub-detection. Since a lot of pixels are affected by zipper effect, the rate of sub-detected edge pixels is generally lower than that of over-detected ones.

Isolated pixels affected by false colors do not always change the location of detected edge pixels. Figure 41 shows that pixels affected with false colors do not change the quality of edge detection, on an extract of the image “Houses”. At these pixels indeed, the gradient module increases, whereas the location of edge pixels remains unchanged.

On the other hand, when the local density of pixels affected by false colors is high, they cause edge over-detection. In textured areas with thin details, most demosaicing schemes generate a lot of neighboring pixels affected by false colors. The gradient module at these pixels increases since its computation takes into account several neighboring false colors. The gradient module at local maxima increases, so that it may become higher than the high threshold T_h used by the hysteresis thresholding. In that case, new edge pixels are detected. For example, figure 40 shows that edge pixels are over-detected in textured areas which correspond to the shutters and to the tiles of the house roofs.

Finally, we notice that statistics about sub-detected edge pixels can be exploited to measure the blurring effect caused by demosaicing, and that over-detected pixels are located in areas with a high density of false colors.

6. Conclusion

This paper is related to the majority of digital color cameras, which are equipped with a single sensor. The surface of this sensor is covered by a color filter array which consists in a mosaic of spectrally selective filters, so that each sensor element samples only one of the three color components Red, Green or Blue. We focus on the Bayer CFA which is the most widely used. To estimate the color (R,G,B) of each pixel in a true color image, one has to determine the values of the two missing color components at each pixel in the CFA image. This process is commonly referred to as *CFA demosaicing*, and its result as the *demosaiced image*.

Demosaicing methods may exploit the spatial and/or frequency domains. The spatial domain has been historically used first, and many methods are based on assumptions about spectral and/or spatial correlation. More recently, works have appeared that exploit the frequency domain, which opens wide perspectives.

We have compared the performances reached by ten demosaicing schemes applied to twelve images extracted from Kodak database, with respect to three kinds of quality measurements : classical fidelity criteria, artifact-sensitive measurements and measurements dedicated to edge detection. The rankings between the demosaicing schemes established thanks to these measurements are consistent.

This detailed evaluation highlights that the methods which primarily analyze the frequency domain outperform those which only scan the spatial domain. More precisely, the methods proposed by Dubois (2005) and by Lian et al. (2007) provide the best demosaicing results whatever the criterion used.

The implementation of demosaicing schemes has to respect real-time constraints. Indeed, the time required for image demosaicing has to be lower than the image acquisition time. Hence, it would be useful to look for a compromise between the processing time and the performance reached by the examined demosaicing schemes. This study would allow to select the best methods which are less time-consuming.

Thanks to a visual comparison of the results, we have described the relationships between artifacts and edge detection quality. Zipper effect causes edge sub-detection, whereas a high density of pixels affected with false colors tends to cause over-detection of edge pixels. These preliminary conclusions are worth being generalized to the relationships between artifacts and the detection quality of other features in the demosaiced images.

Alleysson, D., Chaix de Lavarène, B., Süsstrunk, S., Hérault, J., Sep. 2008. Linear Minimum Mean Square Error Demosaicking. CRC Press, Ch. 8, pp. 213–237.

Alleysson, D., Süsstrunk, S., Hérault, J., Apr. 2005. Linear demosaicing inspired by the human visual system. IEEE Transactions on Image Processing 14 (4), 439–449.

Astola, J., Haavisto, P., Neuvo, Y., Apr. 1990. Vector median filters. Proceedings of the IEEE 78 (4), 678–689.

Asuni, N., Giachetti, A., Jan. 2008. Accuracy improvements and artifacts removal in edge-based image interpolation. In : Ranchordas, A., Araújo, H. (Eds.), Proceedings of the 3rd International Conference on Computer Vision Theory and Application (VISAPP'08). Funchal, Madeira, Portugal, pp. 58–65.

Bayer, B. E., Jul. 1976. Color imaging array. U.S. patent 3,971,065, to Eastman Kodak Co., Patent and Trademark Office, Washington D.C.

Bayrama, S., Sencar, H. T., Memon, N., Sep. 2008. Classification of digital camera-models based on demosaicing artifacts. Digital Investigation 5 (1-2), 49–59.

Buades, A., Coll, B., Morel, J.-M., Sbert, C., Aug. 2008. Non local demosaicing. In : Proceedings of the 2008 International Workshop on Local and Non-Local Approximation in Image Processing (LNLA'08). Lausanne, Suisse.

Busin, L., Vandenbroucke, N., Macaire, L., 2008. Color spaces and image segmentation. Advances in Imaging and Electron Physics 151, 65–168.

Chang, L., Tan, Y.-P., Jan. 2006. Hybrid color filter array demosaicking for effective artifact suppression. Journal of Electronic Imaging 15 (1), 013003,1–17.

Chen, L., Yap, K.-H., He, Y., Mar. 2008. Subband synthesis for color filter array demosaicking. IEEE Transactions on Systems, Man and Cybernetics 38 (2), 485–492.

Chung, K.-H., Chan, Y.-H., Oct. 2006. Color demosaicing using variance of color differences. IEEE Transactions on Image Processing 15 (10), 2944–2955.

Cok, D. R., Dec. 1986. Signal processing method and apparatus for sampled image signals. U.S. patent 4,630,307, to Eastman Kodak Co., Patent and Trademark Office, Washington D.C.

Cok, D. R., Feb. 1987. Signal processing method and apparatus for producing interpolated chrominance values in a sampled color image signal. U.S. patent 4,642,678, to Eastman Kodak Co., Patent and Trademark Office, Washington D.C.

Cok, D. R., May 1994. Reconstruction of CCD images using template matching. In : Proceedings of the IS&T's 47th Annual Conference, Physics and Chemistry of Imaging Systems (ICPS'94). Vol. 2. Rochester, New York, U.S.A., pp. 380–385.

Condat, L., Nov. 2009. A new random color filter array with good spectral properties. In : Proceedings of the IEEE International Conference on Image Processing (ICIP'09). Cairo, Egypt [to appear].

Deriche, R., May 1987. Using Canny's criteria to derive a recursively implemented optimal edge detector. The International Journal of Computer Vision 1 (2), 167–187.

Di Zenzo, S., Jan. 1986. A note on the gradient of a multi-image. Computer Vision, Graphics, and Image Processing 33 (1), 116–125.

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

Eastman Kodak and various photographers, 1991. Kodak Photo CD PCD0992, Access Software & Photo Sampler, Final version 2.0. [CD-ROM, Part No. 15-1132-01].

Eskicioglu, A. M., Fisher, P. S., Dec. 1995. Image quality measures and their performance. IEEE Transactions on Communications 43 (12), 2959–2965.

Faugeras, O. D., Aug. 1979. Digital color image processing within the framework of a human visual model. IEEE Transactions on Acoustics, Speech, and Signal Processing 27 (4), 380–393.

Freeman, W. T., Dec. 1988. Median filter for reconstructing missing color samples. U.S. patent 4,724,395, to Polaroid Co., Patent and Trademark Office, Washington D.C.

Gribbon, K. T., Bailey, D. G., Jan. 2004. A novel approach to real-time bilinear interpolation. In : Proceedings of the 2nd IEEE International Workshop on Electronic Design, Test and Applications (DELTA'04). Perth, Australia, pp. 126–131.

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

Gunturk, B. K., Glotzbach, J., Altunbasak, Y., Schafer, R. W., Mersereau, R. M., Jan. 2005. Demosaicking : Color filter array interpolation. IEEE Signal Processing Magazine 22 (1), 44–54.

Hamilton, J. F., Adams, J. E., May 1997. Adaptive color plan interpolation in single sensor color electronic camera. U.S. patent 5,629,734, to Eastman Kodak Co., Patent and Trademark Office, Washington D.C.

Hamilton, J. F., Compton, J. T., Feb. 2007. Processing color and panchromatic pixels. U.S. Patent 0,024,879 A1, to Eastman Kodak Co., Patent and Trademark Office, Washington D.C.

Harris, C. J., Stephens, M., Aug. 1988. A combined corner and edge detector. In : Proceedings of the 4th Alvey Vision Conference (AVC'88). Manchester, United Kingdom, pp. 147–151.

Hibbard, R. H., 1995. Apparatus and method for adaptively interpolating a full color image utilizing luminance gradients. U.S. patent 5,382,976, to Eastman Kodak Co., Patent and Trademark Office, Washington D.C.

Hirakawa, K., Sep. 2008. Color filter array image analysis for joint denoising and demosaicking. In : Lukac, R. (Ed.), Single-Sensor Imaging : Methods and Applications for Digital Cameras. CRC Press, pp. 239–261.

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

Jai Corporation, 2000. CV-S3200/S3300 series – Super sensitive DSP color camera (JAI CV-S3300P brochure). http://www.graftek.com/pdf/Brochures/JAI/cv-s3200_3300.pdf.

Kimmel, R., Sep. 1999. Demosaicing : image reconstruction from color CCD samples. IEEE Transactions on Image Processing 8 (9), 1221–1228.

Kröger, R. H. H., Aug. 2004. Anti-aliasing in image recording and display hardware : lessons from nature. Journal of Optics A : Pure and Applied Optics 6, 743Ű748.

Kuno, T., Sugiura, H., Mar. 2006. Imaging apparatus and mobile terminal incorporating same. U.S. Patent 7,019,774 B2, to Mitsubishi Denki Kabushiki Kaisha, Patent and Trademark Office, Washington D.C.

Laroche, C. A., Prescott, M. A., Jun. 1993. Apparatus and method for adaptively interpolating a full color image utilizing chrominance gradients. U.S. patent 5,373,322, to Eastman Kodak Co., Patent and Trademark Office, Washington D.C.

Leitão, J. A., Zhao, M., de Haan, G., Jan. 2003. Content-adaptive video up-scaling for high definition displays. In : Proceedings of the SPIE Conference on Image and Video Communications and Processing (IVCP'03). Santa Clara, California, U.S.A., pp. 612–622.

Li, J. J., Randhawa, S., Sep. 2005. High order extrapolation using Taylor series for color filter array demosaicing. In : Proceedings of the International Conference on Image Analysis and Recognition (ICIAR'05). Vol. 3656 of Lecture Notes in Computer Science. Springer, Berlin-Heidelberg, Toronto, Canada, pp. 703–711.

Li, X., Nov. 2000. Edge directed statistical inference and its applications to image processing. PhD thesis, Princeton University, New Jersey, U.S.A.

Li, X., Mar. 2005. Demosaicing by successive approximation. IEEE Transactions on Image Processing 14 (3), 370–379.

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

Lian, N., Chang, L., Tan, Y.-P., Sep. 2005. Improved color filter array demosaicking by accurate luminance estimation. In : Proceedings of the 12th International Conference on Image Processing (ICIP'2005). Vol. 1. Genova, Italia, pp. I–41–4.

Lian, N.-X., Chang, L., Tan, Y.-P., Zagorodnov, V., Oct. 2007. Adaptive filtering for color filter array demosaicking. IEEE Transactions on Image Processing 16 (10), 2515–2525.

Lian, N.-X., Chang, L., Zagorodnov, V., Tan, Y.-P., Nov. 2006. Reversing demosaicking and compression in color filter array image processing : Performance analysis and modeling. IEEE Transactions on Image Processing 15 (11), 3261–3278.

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

Lu, Y. M., Karzand, M., Vetterli, M., Jan. 2009. Iterative demosaicking accelerated : Theory and fast noniterative implementations. In : Bouman, C. A., Miller, E. L., Pollak, I. (Eds.), Procs. 21th IS&T/SPIE Electronic Imaging Annual Symposium (SPIE'09). Vol. 7246 of Computational Imaging VII. San Jose, California, USA, pp. 72460L–72460L–12.

Lukac, R., Sep. 2008. Single-Sensor Imaging : Methods and Applications for Digital Cameras. Image Processing Series. CRC Press / Taylor & Francis, Boca Raton, Florida, USA.

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

Lukac, R., Plataniotis, K. N., Oct. 2004b. A normalized model for color-ratio based demosaicking schemes. In : Proceedings of the 11th International Conference on Image Processing (ICIP'04). Singapore, pp. 1657–1660.

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

Lukac, R., Plataniotis, K. N., Apr. 2005b. Universal demosaicking for imaging pipelines with an RGB color filter array. Pattern Recognition 38, 2208–2212.

Lukac, R., Plataniotis, K. N., 2007. Single-sensor camera image processing. In : Lukac, R., Plataniotis, K. N. (Eds.), Color Image Processing : Methods and Applications. CRC Press / Taylor & Francis, pp. 363–392.

Lukac, R., Plataniotis, K. N., Hatzinakos, D., Aleksic, M., Jul. 2006. A new CFA interpolation framework. Signal Processing 86 (7), 1559–1579.

Lukin, A., Kubasov, D., Sep. 2004. An improved demosaicing algorithm. In : Proceedings of the 14th International Conference on Computer Graphics, GRAPHICON'04. Moscow, Russia, pp. 38–45.

Lyon, R., Hubel, P. M., Nov. 2002. Eyeing the camera : into the next century. In : Proceedings of the 10th Color Imaging Conference (CIC'2002) : Color Science and Engineering Systems, Technologies, Applications. Scottsdale, Arizona, U.S.A., pp. 349–355.

Lyon, R. F., Mar. 2000. Prism-based color separation for professional digital photography. In : Proceedings of the IS&T Conference on Image Processing, Image Quality, Image Capture, Systems (PICS'00). Vol. 3. Portland, Oregon, USA, pp. 50–54.

Martin, D. R., Fowlkes, C., Malik, J., May 2004. Learning to detect natural image boundaries using local brightness, color, and texture cues. IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (5), 530–549.

Marziliano, P., Dufaux, F., Winkler, S., Ebrahimi, T., 2004. Perceptual blur and ringing metrics : application to JPEG2000. Signal Processing : Image Communication 19, 163–172.

Medjeldi, T., Horé, A., Ziou, D., Jul. 2009. Enhancement of the quality of images through complex mosaic configurations. In : Kamel, M., Campilho, A. (Eds.), Proceedings of the International Conference on Image Analysis and Recognition (ICIAR'09). Vol. 5627. Springer, Berlin-Heidelberg, Halifax, Canada, pp. 43–53.

Menon, D., Andriani, S., Calvagno, G., Sep. 2006. A novel technique for reducing demosaicing artifacts. In : Proceedings of the XIVth European Signal Processing Conference (EUSIPCO'06). Firenze, Italia.

Menon, D., Andriani, S., Calvagno, G., Jan. 2007. Demosaicing with directional filtering and *a posteriori* decision. IEEE Transactions on Image Processing 16 (1), 132–141.

Mukherjee, J., Parthasarathi, R., Goyal, S., Mar. 2001. Markov random field processing for color demosaicing. Pattern Recognition Letters 22 (3-4), 339–351.

Muresan, D. D., Luke, S., Parks, T. W., Aug. 2000. Reconstruction of color images from CCD arrays. In : Proceedings of the Texas Instruments Digital Signal Processing Systems Fest. Houston, Texas, U.S.A., pp. 1–6 [CD–ROM XP002243635].

Noble, S. A., 2000. The technology inside the new Kodak Professional DCS 620x digital camera. http://www.dpreview.com/news/0005/kodak_dcs620x_tech_paper.pdf.

Omer, I., Werman, M., Oct. 2004. Using natural image properties as demosaicing hints. In : Proceedings of the 11th International Conference on Image Processing (ICIP'04). Vol. 3. Singapore, pp. 1665– 1670.

Parulski, K. A., Aug. 1985. Color filters and processing alternatives for one-chip cameras. IEEE Transactions on Electron Devices 32 (8), 1381–1389.

Roorda, A., Metha, A. B., Lennie, P., Williams, D. R., Jan. 2001. Packing arrangement of the three cone classes in primate retina. Vision Research 41, 1291–1306.

Savard, J., 2007. Color filter array designs [online]. http://www.quadibloc.com/other/cfaint.htm.

Smith, M., Apr. 2005. Super-resolution. Tech. rep., Carleton University, Carleton, Canada.

Sony Corporation, 2000. Diagonal 6mm (type 1/3) CCD image sensor for NTSC color video cameras (ICX258AK) (JAI CV-S3300P datasheet). http://www.jai.com/SiteCollectionDocuments/Camera_Solutions_Other_Documents/ICX258AK.pdf.

Su, C.-Y., May 2006. Highly effective iterative demosaicing using weighted-edge and color-difference interpolations. IEEE Transactions on Consumer Electronics 52 (2), 639–645.

Su, D., Willis, P., Jun. 2003. Demosaicing of colour images using pixel level data-dependent triangulation. In : Proceedings of the Theory and Practice of Computer Graphics (TPCG'03). Birmingham, United Kingdom, pp. 16–23.

Tam, W.-S., Kok, C.-W., Siu, W.-C., Aug. 2009. A modified edge directed interpolation for images. In : Proceedings of the 17th European Signal Processing Conference (EUSIPCO'09). Glasgow, Scotland.

Tsai, C.-Y., Song, K.-T., Sep. 2007. A new edge-adaptive demosaicing algorithm for color filter arrays. Image and Vision Computing 25 (9), 1495–1508.

Wang, Z., Bovik, A. C., 2006. Modern Image Quality Assessment. Synthesis Lectures on Image, Video, and Multimedia Processing. Morgan & Claypool Publishers.

Wang, Z., Bovik, A. C., Jan. 2009. Mean squared error : Love it or leave it ? – A new look at signal fidelity measures. IEEE Signal Processing Magazine 26 (1), 98–117.

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

Yang, Y., Losson, O., Duvieubourg, L., Dec. 2007. Quality evaluation of color demosaicing according to image resolution. In : Proceedings of the 3rd International Conference on Signal-Image Technology & Internet-based Systems (SITIS'07). Shanghai Jiaotong University, China, pp. 689–695.

Zapryanov, G. S., Nikolova, I. N., Nov. 2009. Demosaicing methods for pseudo-random Bayer color filter array. In : Proceedings of the 5th International Conference – Computer Science'09 (CS'09). Sofia, Bulgaria, pp. 687–692.

Zhang, X., Wandell, B. A., 1997. A spatial extension of CIELAB for digital color reproduction. Journal of the Society for Information Display 5 (1), 61–63.