

Recent Patents on Color Demosaicing

Sebastiano Battiato^{1,*}, Mirko Ignazio Guarnera², Giuseppe Messina^{1,2} and Valeria Tomaselli²

¹*Dipartimento di Matematica ed Informatica, Università di Catania, Italy,* ²*STMicroelectronics, Bld. M5, Stradale Primosole, Catania, Italy*

Received: May 22, 2007; Accepted: May 26, 2008; Revised: June 30, 2008

Abstract: Single-sensor technology is a popular imaging approach used in image-enabled consumer electronic devices such as digital still and video cameras, mobile phones, personal digital assistants, and visual sensors for surveillance and automotive applications. Cameras make use of an electronic sensor (Charge Coupled Device - CCD - or Complementary Metal-Oxide-Semiconductor - CMOS) to acquire the spatial variations in light intensity and then use image processing algorithms to reconstruct a color picture from the data provided by the sensor. Acquisition of color images requires the presence of different sensors for different color channels. Manufacturers reduce the cost and complexity by placing a color filter array (CFA) on top of a single image sensor, which is basically a monochromatic device, to acquire color information of the true visual scene. Typical imaging pipelines implemented in single-sensor cameras are usually designed to find a trade-off between sub-optimal solutions (devoted to solve imaging acquisition) and technological problems (e.g., color balancing, thermal noise, etc.) in a context of limited hardware resources. In this paper we review the existing patent solutions devoted to demosaicing and able to generate a color image from a single-sensor reading. Demosaicing solutions can be basically divided into four main categories: inter-channel (spectral) correlation, edge based, pattern based and iterative together with alternative techniques also present in literature. Discussion about pro and cons of each technique will be briefly reported.

Keywords: Color demosaicing, color interpolation, spatial correlation, spectral correlation, antialiasing, recent patents.

INTRODUCTION

Imaging Devices (Digital Cameras, PDA, Mobile Phones, etc.) are becoming more and more ubiquitous, replacing de facto the film-based camera in all camera based applications. To reduce the costs and size, typical devices use the expedient to capture the image using only one sensor chip (CCD or CMOS), covering its surface with a Color Filter Array (CFA). The CFA is compound by a set of spectrally selective filters, arranged in an interleaved mosaic pattern, so that each pixel registers only one of the components of the color spectrum.

B.E. Bayer proposed in the 1975 a Pattern [1] referred as Bayer Pattern (Fig. (1)), based on the principle that the human eye is more sensitive to the luminance than chrominance; thus the luminance (green) has to be sampled at higher rate than the chrominance channels. This is due to the Green channel response curve that is quite fit to the luminance one (around 550nm).

Other patterns have been proposed with different arrangements of RGB colors (as shown in Fig. 2.a), or with different color sets. Three complementary colors CMY and four colors systems (RGB plus a fourth component, such as white or emerald) are alternative to classic RGB patterns (as shown in Fig. (2.b)).

However, most of the patents on demosaicing exploit the Bayer arrangement.

Since only one primary color for each pixel is available, color interpolation techniques, using the neighboring pixels, must be used to reconstruct the missing color information at

each pixel location. These methods are commonly referred as Demosaicing or Color Interpolation algorithms and have a large influence on the final image quality. Demosaicing is one of the fundamental steps in the Image Generation Pipeline (IGP) for any single-sensor imaging devices.

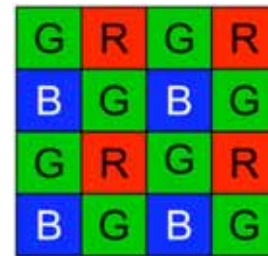


Fig. (1). Bayer CFA pattern.

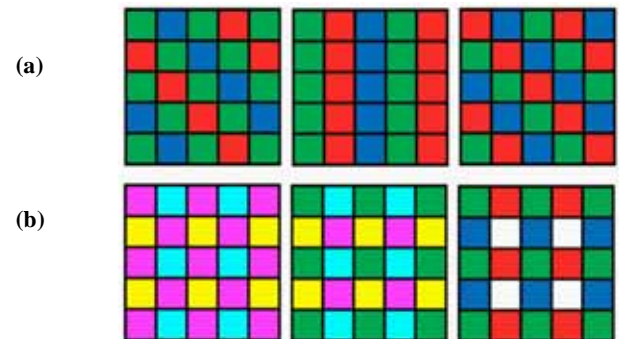


Fig. (2). Other CFA arrangements. (a) Represents different pattern arrangements based on RGB; (b) represents patterns based on CMY, mixed version, and four channel version (in this case the fourth channel is white).

*Address correspondence to this author at the Dipartimento di Matematica ed Informatica, Università di Catania, Italy; E-mail: battiato@dmi.unict.it

The IGP usually consists of a Preprocessing Block (Auto-Focus, Auto-Exposure, etc.), a Color Interpolation, a White Balancing, a Color Matrixing step (that corrects the colors depending on the sensor architecture), and Post-processing block (sharpening, etc.). A typical pipeline, that creates an image to be stored or compressed, is shown in Fig. (3).

A huge number of demosaicing algorithms have been patented. In the following section we will outline some of the most known and recent methods.

Before going into details, some initial considerations must be appointed. Given an input image $I(x,y)$, let $I(m,n)$ be its CFA image. From the sampling theory it is well known that the Fourier transform of the sampled image $I(m,n)$ contains scaled, periodic replications of the Fourier transform of the original image $I(x,y)$ [2]. If the sampling is ideal, the repetitions do not overlap each other, thus the image $I(x,y)$ can be faithfully reconstructed, otherwise the aliasing phenomenon occurs. The non-overlapping constraint implies a limited band, thus a not limited space, which can not be performed. For this reason, in real cases, aliasing effects appear. The aliasing (Fig. (4)) arises with false patterns or colors, and happens when the image frequencies exceed the sampling one.

The green channel is less affected by aliasing than the red and blue channels, because it is sampled at a higher rate. Although demosaicing solutions aim to eliminate, limit or correct false colors and other impairments caused by a non ideal sampling, residual artifacts can still remain. That is why some specific solutions (and patents) apply ad-hoc filter in a post-processing phase, as described in Section 3. Post-processing techniques are usually more powerful in achieving false colors removal and sharpness enhancement, because their inputs are fully restored RGB color images. Moreover, they can be applied more than once, in order to meet some quality criteria.

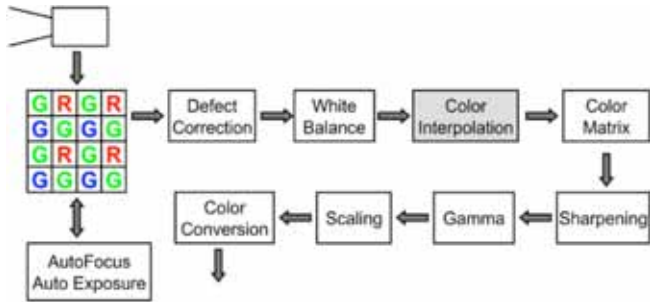


Fig. (3). An Image Generation Pipeline (IGP) of a typical imaging device.

The simplest class of demosaicing methods interpolates each channel separately. The bilinear interpolation, which belongs to this class, is the most widely used due to its simplicity. Referring to Sakamoto [3], interpolation in mathematical terms can be formally expressed as:

$$c(x,y) = \sum_k \sum_l c(x_k, y_l) h(|x - x_k|) h(|y - y_l|) \quad (1)$$

where $c(x,y)$ is the interpolated value, h is the interpolation kernel and $c(x_k, y_l)$ is the sampled value at pixel (x_k, y_l) . The bilinear interpolation kernel is simply defined as:

$$h(x) = \begin{cases} 1-x & 0 \leq x \leq 1 \\ 0 & 1 < x \end{cases} \quad (2)$$

The missing channels are computed as the weighted averages of their neighboring pixel values. This method is computationally light and very easy to implement, but its band-limiting (Low Pass Filtered - LPF) nature smooths edges and highlights well-known zipper effects. The zipper effect (Fig. (5)) is revealed when interpolation is achieved across edges, and in areas with sudden jumps from low to high frequencies. The edges will then look like a zipper or colored fringes.

Hundred of papers on demosaicing solution have also been published in the last years, exploiting a lot of different approaches. An extensive dissertation of these approaches can be found in [4].

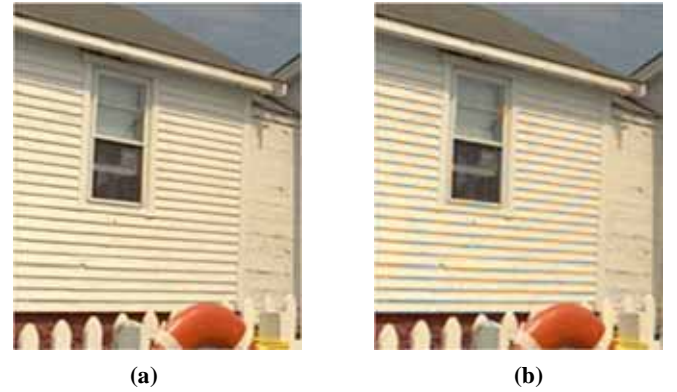


Fig. (4). Aliasing effect. (a) a particular of the original image, while (b) is the same particular in the interpolated image, where false colors occur.

Among the most recent solutions we mention the paper in [5], where the authors propose an algorithm which classifies direction for interpolation according to both intra-channel differences and inter-channel differences, taking also into account neighboring interpolation direction. This information consolidates the process of selecting correct directions before interpolation. Moreover, in order to keep consistency along edge, refinements are made on pixels that have ambiguous situations by using neighbors' categories to classify pixels without category. In [6] the properties of the spectrum of the CFA image are analyzed and, using suitable filters, the luminance of the image is reconstructed, and then an estimate of the full-color image is obtained. In [7] a wavelet-based analysis of the luminance component is performed to drive an adaptive interpolation algorithm of the color components of the image. [8] reports that the high-frequency contents between different color planes are strongly correlated by calculating the correlation values between the detail (high-frequency) wavelet coefficients. This method achieves good performance and hence it is a popular benchmark algorithm in CFA research literature and also in some patents. Based on the observation that the luminance is well separated from the chrominance in the

spatial frequency domain of a Bayer CFA image, in [9] a frequency selection approach to preserving high-frequency information is proposed; here the chrominance is the image color-difference component. The authors use a low-pass filter to reconstruct from a CFA image a full-resolution luminance plane, which is then used as a reference to reconstruct the missing color values. In [2], through a more detailed analysis of the CFA spectrum, an improvement to the method [9] is proposed. In particular, an efficient low-pass filter for luminance of green samples to better preserve the high-frequency information is used, while the high-frequency information at red/blue pixels is estimated by adaptive filtering of color-difference components.

In [10] the authors disclose a method that can be applied on any mosaic (not only Bayer Pattern), in particular on pseudo-random mosaics. It reveals interesting properties in terms of false colors reduction. The authors starting from spectral model of [9], where the chrominance information can be estimated using simple low-pass filtering, proposed separable recursive filters.

In [11] a learning-based demosaicing is proposed, where a Vector Quantization (VQ)-based method is utilized for learning. The training data is divided into small RGB patches and corresponding degraded mosaic patches are generated. To restore a given mosaic image, it is divided into small patches, that are compared to those of the training data.

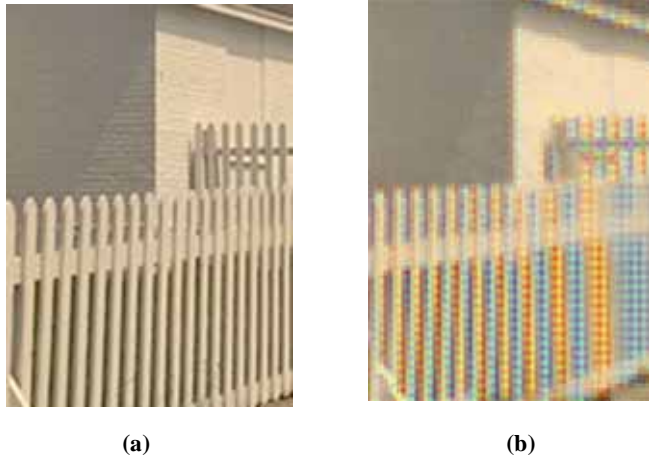


Fig. (5). Zipper effect: (a) a particular of the original image, while (b) is the same particular in the interpolated image, where zipper effects occur.

In this paper we review some recent patented solutions, devoted to demosaicing; such techniques are typically fast and simple to be implemented inside a system with low capabilities (e.g., memory, CPU, low-power, ...). In the following section after a first categorization, the various methods are grouped and presented with respect to their main underlying ideas. Section 3 is devoted to describe the typical antialiasing techniques able to overcome aliasing artifacts. Section 4 briefly reviews the main quality evaluation of interpolated images. Finally current and future developments are disclosed in section 5.

PATENTS CLASSIFICATION

In this section we present different recent patented demosaicing methods, which are classified according to four predefined classes of algorithms: spectral correlation, edge detection (component wise or spatial correlation), pattern recognition, iteration (successive optimizations).

With reference to Fig. (6), the methods exploiting spatial correlation process each channel of the color image separately (see Fig. (6a)). According to this principle, within a homogeneous image region, neighboring pixels share similar color values, so a missing value can be retrieved by averaging the pixels close to it. On the other hand techniques which are based on spectral correlation use the information coming from all the color channels to interpolate each color channel, as shown in Fig. (6b).

We tried to associate each patent proposal to a single class, but it is not always possible. In fact a lot of methods exploit more than one approach, becoming a kind of mixture of different techniques. We present also some patents based on alternative approaches (sometime unfeasible due to high HW requirements), which cannot be assigned to any of the aforementioned classes.

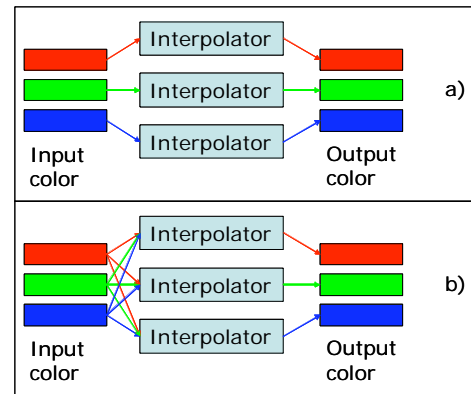


Fig. (6). Image processing paradigms: (a) spatial processing; (b) spectral processing.

Spectral Correlation

In this class of algorithms final RGB values are derived taking into consideration the inter-channel color correlations in a limited region.

As already mentioned, Gunturk *et al.* in [8] have demonstrated that high frequency components of the three color planes are highly correlated, but not equal. This suggests that any color component can help to reconstruct the high frequencies for the remaining color components.

For instance, if it is assumed a red central pixel, the green component can be determined as:

$$G(i,j) = G_{LPF}(i,j) + R_{HPF}(i,j) \quad (3)$$

where $R_{HPF}(i,j) = R(i,j) - R_{LPF}(i,j)$ is the High Frequency content of the R channel, and G_{LPF} and R_{LPF} are the Low Frequency components of the G and R channels, respectively.

This implies that the green channel can take advantage of the red and blue information.

Furthermore for real world images the color difference planes ($\Delta_{GR}=G-R$ and $\Delta_{GB}=G-B$) are rather flat over small regions, and this property is widely exploited in demosaicing and antialiasing techniques.

For example, some techniques median filter the color difference values in order to make pixels more similar to their neighbors, thus reducing false colors. This model using channel differences (that can be viewed as chromatic information), is closer to the Human Color Vision system that is more sensitive to chromatic changes than luminance changes in low spatial frequency regions. Like the previous example, if the central pixel is R, the green component can be derived as:

$$G(i,j) = R(i,j) + \Delta_{GR} \quad (4)$$

The method proposed in [12] belongs to this class. This technique generates by first a LPF version of each one of the three channels (R , G and B) by taking into consideration an edge strength metric to inhibit smoothing of detected edges. Then a difference between the estimated smoothed values and the original Bayer pattern values is performed to obtain the corresponding High Frequency values. Finally the LPF channels and the corresponding estimated HPF planes are combined into the final RGB image.

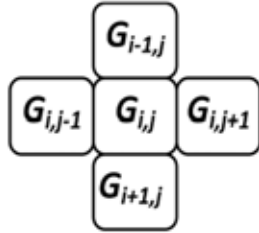


Fig. (7). Pattern of five pixels used to calculate an edge metric on a central G pixel of the LPF G image.

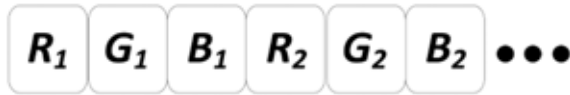


Fig. (8). Interleaved RGB stripe pattern.

In particular the HPF values are obtained through the following relations:

Table 1. Color Correlation Defined in [12]

	At a Red Pixel	At a Green Pixel	At a Blue Pixel
R	R	$R_{LPF} + G - G_{LPF}$	$R_{LPF} + B - B_{LPF}$
G	$G_{LPF} + R - R_{LPF}$	G	$G_{LPF} + B - B_{LPF}$
B	$B_{LPF} + R - R_{LPF}$	$B_{LPF} + G - G_{LPF}$	B

Each smoothed LPF image is formed by a two-dimensional interpolation combined with a low-pass filtering

excepted for pixels that maximize the edge strength metric. For example, if the central pixel is a G pixel the four adjacent G pixels, which will be taken into consideration to estimate the edge strength, are generated by interpolation (Fig. (7)).

Thus the measure of edge strength E_{ij} , that is proportional to the square of the actual edge difference, is then calculated according to:

$$E_{ij} = (G_{ij} - G_{i,j-1})^2 + (G_{ij} - G_{i,j+1})^2 + (G_{ij} - G_{i-1,j})^2 + (G_{ij} - G_{i+1,j})^2 \quad (5)$$

By considering this edge metric the algorithm reduces the presence of color artifacts on edges boundaries.

Another color correlation algorithm is also presented in [13] but this approach is based on a different type of Color Filter Array (not Bayer pattern). Such technique is applied on an interleaved RGB Stripe Pattern (Fig. (8)).

This patent considers a linear interpolation to obtain the three channels. Assuming R_1 and R_2 as original sampled data,

$$r'_1 = \frac{2}{3}R_1 + \frac{1}{3}R_2 \quad \text{and} \quad r''_1 = \frac{1}{3}R_1 + \frac{2}{3}R_2 \quad (6)$$

will be the missing red values for the pixels G_1 and B_1 . To avoid color artifacts a further correction is performed using the spectral correlation among estimated and real R , G , and B values. The differences $R-G$ and $B-G$ are filtered by a horizontal median filter of 9 pixels, obtaining the new values $(R-G)'$ and $(B-G)'$. The colors are then restored using the relationships in Table 2:

Table 2. Color Correlation Defined in [13]

	At a Red Pixel	At a Green Pixel	At a Blue Pixel
R	R	$G + (R-G)'$	$B - (B-G)' + (R-G)'$
G	$R - (R-G)'$	G	$B - (B-G)'$
B	$R - (R-G)' + (B-G)'$	$G + (B-G)'$	B

It is obvious that this method belongs to the general class of eq. (4). However the main problem of such approach is the total independence with edges, thus it would be possible to find false colors along the diagonals.

The method [14] uses an adaptive interpolation technique for each type of Bayer Pattern pixel (R , B , green in the red row G_R and green in the blue row G_B). In particular five different interpolators are considered. Generally, to generate estimated values very close to actual pixel values, it applies a nonlinear low pass filter (NLPF) that reflects the change rate of the data around the center pixel and the data of the central pixel, and by simultaneously applying a low pass filter (LPF), a band pass filter (BPF) and high pass filter (HPF) having linear characteristics, thus reducing aliasing and emphasizing high frequencies. As the process of the interpolation is strictly related to the local position on the Bayer pattern, the Table 3 is introduced as summary of the approach:

Table 3. Color Correlation Defined in [14]

	Center R	Center B	Center G _r	Center G _b
R		<i>Eq.CC.C2</i>	<i>Eq.CC.C4</i>	<i>Eq.CC.C4</i>
G	<i>Eq.CC.C1</i>	<i>Eq.CC.C1</i>		
B	<i>Eq.CC.C2</i>		<i>Eq.CC.C3</i>	<i>Eq.CC.C3</i>

Where the equations to be taken into consideration are:

Eq.CC.C1

$$C_{y,x} = \frac{(a_1 C_{y-1,x} + a_3 C_{y,x+1} + a_5 C_{y+1,x} + a_7 C_{y,x-1})}{(a_1 + a_3 + a_5 + a_7)}$$

Eq.CC.C2

$$C_{y,x} = \frac{(a_2 C_{y-1,x+1} + a_4 C_{y+1,x+1} + a_6 C_{y+1,x-1} + a_8 C_{y-1,x-1})}{(a_2 + a_4 + a_6 + a_8)}$$

Eq.CC.C3

$$C_{y,x} = \frac{(a_1 C_{y-1,x} + a_5 C_{y+1,x})}{(a_1 + a_5)}$$

Eq.CC.C4

$$C_{y,x} = \frac{(a_3 C_{y,x+1} + a_7 C_{y,x-1})}{(a_3 + a_7)}$$

where the coefficients a_i , with $i=1,...,8$, are weighting factor estimated through the distance among the central $C_{y,x}$ pixel and the surrounding values in a window of 5x5 pixels. These weights depend on both the channel to which the central pixel belong and the channel to be interpolated.

The usage of LPF, BPF and HPF in conjunction to the Non linear Low pass Filter allows to reduce aliasing (at “edges”) and emphasize the high frequencies components. This invention assumes that original signals (i.e., G , R) and linear LPF signals (i.e., G_{LPF} , R_{LPF}) have almost the same difference ratios on the identical locations, thus exploiting spectral correlation.

In [15] a method based on the smooth hue transition algorithms by using the color ratio rule is proposed. This rule is derived from the photometric image formation model, which assumes the color ratio is constant in an object. Each color channel is composed of the Albedo multiplied by the projection of the surface normal onto the light source direction. The Albedo is the fraction of incident light that is reflected by the surface, and is function of the wavelength (is different for each color channel) in a Lambertian surface (or even a more complicate Mondrian). The Albedo is constant in a surface, then the color channel ratio is hold true within the object region.

This class of algorithms, instead of using inter-channel differences, calculates the green channel using a well-known interpolation algorithm (i.e., bilinear or bicubic), and then computes the other channels using the red to green and blue to green ratios, defined as:

$$H_b = \frac{B}{G} \quad \text{and} \quad H_r = \frac{R}{G}. \quad (7)$$

An example of such method is described in [16]. In this patent the Bayer data are properly processed by a LPF circuit and an adaptive interpolation module.

The LPF module cuts off the higher frequency components of the respective color signals R , G and B and supplies R_{LPF} , G_{LPF} , and B_{LPF} . On the other hand, the adaptive interpolation circuit calculates a local pixel correlation from the color signals R and G and executes interpolation with a pixel which maximizes the correlation to obtain a high resolution luminance signal.

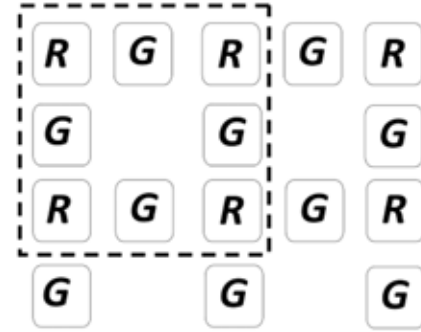


Fig. (9). RG pixel map for luminance interpolation.

The authors assume that, since the color signals R and G have been adjusted by the white balance module, they have almost identical signal levels, so they can be considered as luminance signals. So, taking into consideration the Bayer pattern selected in Fig. (9), they assume that luminance signals are arranged as shown in Fig. (10), where the value Y_5 has to be calculated according to the surrounding values.

The correlation S can be defined for a pixel string Y_n of a particular direction as follows (similarly to eq. (7)):

$$S = \min(Y_n) / \max(Y_n) \quad (8)$$

where $S \leq 1$ and the maximum correlation is obtained when $S=1$.

The correlation is calculated for the horizontal, vertical and diagonal directions, and interpolation is executed in a direction which maximizes the correlation. For instance, the correlation value a in the vertical direction is calculated as follows:

$$a = \min(Y_n) / \max(Y_n) \quad (9)$$

where

$$\min(Y_n) = \min(Y_1, Y_4, Y_7) \cdot \min(Y_2, Y_8) \cdot \min(Y_3, Y_6, Y_9) \quad (10)$$

and

$$\max(Y_n) = \max(Y_1, Y_4, Y_7) \cdot \max(Y_2, Y_8) \cdot \max(Y_3, Y_6, Y_9). \quad (11)$$

The correlations in the horizontal and diagonal directions are computed in a similar way.

If the direction which maximizes the correlation is the vertical one, the interpolation is executed as follows:

$$Y_5 = (Y_2 + Y_8) / 2. \quad (12)$$

Another way to decide the direction is to consider the similarities between the pixels. The dispersion degree of the color R is calculated as:

$$\sigma R = \min(R_1, R_2, R_3, R_4) / \max(R_1, R_2, R_3, R_4). \quad (13)$$

If the dispersion degree is greater than a threshold, interpolation along a diagonal direction is executed. On the contrary, when the dispersion degree is small, correlation of the color R is almost identical in any directions, so it is possible to interpolate only G

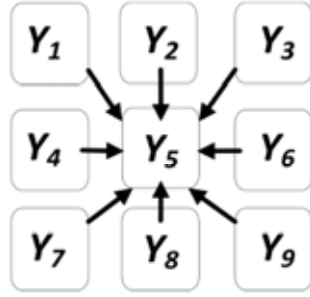


Fig. (10). Luminance map of analyzed signal.

along the vertical or horizontal direction. This means that the interpolation is executed only with G having the highest frequency, thus enabling to obtain an image of a higher resolution.

Once the luminance signal Y is interpolated, an HPF module is provided with the luminance signal Y and the color signals R and G . The HPF module creates a luminance signal Y_{HPF} of a higher frequency component. Finally, an adder combines the lower frequency components of the respective color signals R_{LPF} , G_{LPF} , and B_{LPF} with the higher frequency component luminance signal Y_{HPF} and outputs color signals R , G and B of a high resolution.

A brief mention to the specific application described in [17] that considers the correlation among the three channels to estimate the presence of gray level and prevent the color interpolation step using a simpler hue based interpolation.

EDGE Based

One of the principles of color interpolation techniques is to exploit spatial correlation. Edge based algorithms exploit the spatial correlation principle by interpolating along edges and not across them. Techniques which disregard directional information often produce images with zipper effect. On the contrary, techniques which interpolate along edges are less affected by this kind of artifact. Furthermore, averaging of pixels which are across an edge also leads to a decrease in the sharpness of the image at the edges.

Edge based color interpolation techniques are widely disclosed in literature, and they can be differentiated mainly according to the number of directions, the way adopted to choose the direction and the interpolation method.

The method in [18] discloses a technique which firstly interpolates the green color plane, then interpolates the remaining two planes. A missing G pixel can be interpolated horizontally, vertically or by using all the four samples

around it. With reference to the neighborhood of Fig. (11) the interpolation direction is chosen through two classifiers

$$\begin{aligned} \Delta H &= -A3 + 2A5 - A7 + G4 - G6 \\ \Delta V &= -A1 + 2A5 - A9 + G2 - G8 \end{aligned} \quad (14)$$

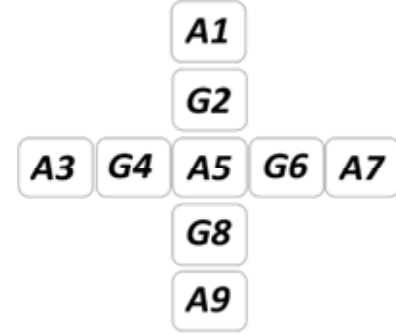


Fig. (11). Considered Neighborhood.

which are composed of Laplacian second-order terms for the chroma data and gradients for the green data.

Once the G color plane is interpolated, R and B at G locations are interpolated. In particular, a horizontal predictor is used if their nearest neighbors are in the same row, whereas a vertical predictor is used if their nearest neighbors are in the same column. Finally, R is interpolated at B locations and B is interpolated at R locations.

Since their nearest neighbors are placed in the corners of the 3×3 neighborhood, the interpolation direction can be negative diagonal, positive diagonal or an average of the four values. It is chosen through two classifiers which are composed of Laplacian second-order terms for the green data and gradients for the chroma data. The classifiers used in this invention have the advantage of being inflated by high spatial frequency information in either the green data or the chroma data in a certain direction. The interpolation is achieved by averaging the neighboring pixels of the missing channel along the identified direction, and adding a Laplacian correction term of another color channel, which may contain edge information not available in the first color channel. Although the interpolation is not just an average of the neighboring pixels, wrong colors can be introduced near edges, especially when the luminance of the image changes less or more than an individual color. To improve the performance of this invention, the patent [19] introduces the possibility to control the Laplacian correction term. This control mechanism allows increasing the sharpness of the image, reducing at the same time wrong colors and ringing effects near edges. In particular, if the Laplacian correction term is greater than a predefined threshold, it is changed by calculating an attenuating gain for it, which depends on the minimum and maximum values of the G channel and of another color channel. A drawback of these two inventions is that G can be interpolated only in horizontal and vertical directions; R and B can be interpolated only in diagonal directions (B and R central pixel) or in horizontal and vertical directions (G central pixel).

The patent [20], similarly to the previous two, interpolates the missing green values in either

$$\begin{array}{cc}
 [1 & -1 & -2 & 1 & 1] & \begin{bmatrix} 1 \\ -1 \\ -2 \\ 1 \\ 1 \end{bmatrix} \\
 \text{Horizontal mask} & \text{Vertical mask}
 \end{array}$$

Fig. (12). Variation Masks proposed in [20].

horizontal or vertical direction, and chooses the direction depending on the intensity variations within the observation window. The variation filters, shown in Fig. (12), take into account both G and non- G intensity values. From Fig. (12) it is readily apparent that these variation filters are quite similar to the classifiers of [18]. The main difference is the absence of the absolute values. In this case, the interpolation of G values is achieved through a simple average of the neighboring pixels in the chosen direction, but the quality of the image is improved by applying a sharpening filter.

One important feature of this invention is the G_R-G_B mismatch compensator block (where G_R is green in the red row and G_B is green in the blue row), which tries to overcome the green imbalance issue. In some sensors the photosensitive elements that capture G intensity values at G_R locations can have a different response than the photosensitive elements that capture G intensity values at G_B locations. The G_R-G_B mismatch module applies gradient filters and curvature filters to derive the maximum variation magnitude. If this value exceeds a predefined threshold value, the G_R-G_B smoothed intensity value is selected, otherwise the original G intensity value is selected. To interpolate the missing R and B values, the spectral correlation is exploited. In fact, discontinuities of all the color components are assumed to be equal. Thus, color discontinuity equalization is achieved by equating the discontinuities of the remaining color components with the discontinuities of the green color component. Methods which use inter-channel correlation in addition to edge estimation usually provide higher quality images.

The patent [21] firstly interpolates the G color plane, by distinguishing among three different cases: horizontal edge, vertical edge, no edges. In this case, edges are detected by comparing each of two G reference pixels placed at the top and bottom or left and right of the target position with four secondary reference pixels of the same color, existing in the vicinity of the first reference pixels. Thus, the information from the other color channels is not taken into account.

With reference to Fig. (13), for example, a horizontal edge is detected if both the pixels at the left and right of the target position have their intensity values lower (or higher) than the 6 secondary reference pixels. When an image edge is detected, the G missing color value is interpolated based on reference G pixels existing in adjacent positions at the top, bottom, left and right of the target position. Once the green color values at all positions are available, they are used in the interpolation of the R and B color channels. In particular, a linear interpolation by the chrominance signals is achieved. One benefit of this invention is its simplicity, so it can be easily implemented in hardware, but, on the other

hand, the detection of the edge directions can be prone to errors in case of noisy images.

All the already disclosed patents propose an adaptive interpolation process in which some conditions are evaluated to decide between the horizontal and vertical interpolation. When neither a horizontal edge nor a vertical edge is identified, the interpolation is performed using an average value among surrounding pixels. This means that resolution in appearance deteriorates in the diagonal direction. Moreover, in regions near the vertical and horizontal Nyquist frequencies, the interpolation direction can abruptly change, thus resulting in unnaturalness in image quality. To overcome the above mentioned problems, the patent [22] prevents an interpolation result from being changed discontinuously with a change in the correlation direction. First of all, vertical (ΔV) and horizontal correlation values (ΔH) of a target pixel to be interpolated are calculated by using the equations in (14) (like the patent [18]). Then, a coefficient value depending on the direction in which the target pixel has higher correlation is computed, according to the following formula:

$$K = \begin{cases} 0 & \text{if } \Delta H = \Delta V \\ 1 - \frac{\Delta V}{\Delta H} & \text{if } \Delta H > \Delta V \\ \frac{\Delta H}{\Delta V} - 1 & \text{if } \Delta H < \Delta V \end{cases} \quad (15)$$

thus K has a value from -1 to 1.

The coefficient value is used to weight the interpolation data in the vertical or horizontal direction with the interpolation data in the diagonal direction. If K is a positive value, a weighted average of the vertical interpolated value (V_{value}) and the two-dimensional interpolated value ($2D_{value}$) is calculated from the formula (16), using the absolute value Ka of the correlation coefficient K .

$$\text{Output} = V_{value} \times Ka + 2D_{value} \times (1 - Ka) \quad (16)$$

Obviously, if K is a negative value a weighted average of the horizontal interpolated value and the two-dimensional interpolated value is computed. As a result, a proportion of either the vertical or horizontal direction interpolation data can be continuously changed without causing a discontinuous change in interpolation result when the correlation direction changes.

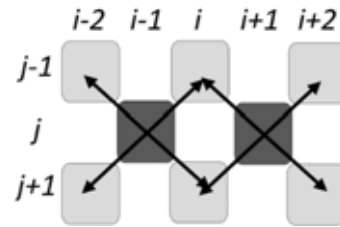


Fig. (13). Example for horizontal edge detection.

The authors of the US patent [23], starting from the consideration that the directivity of an edge is not always symmetrically right and left or symmetrically up and down,

prefer to use 4 different directions (up, down, left and right) instead of simply horizontal and vertical. Moreover some embodiments of this invention use 8 directions (up, down, left, right, up-right, down-right, up-left, down-left). This choice possibly enhances the sharpness of corners, but it could provide images with jaggy edges. The interpolation direction is chosen just analyzing for each Bayer data (*R* or *G* or *B*) neighborhood the preferred direction, by using local oriented differences. The interpolation is obtained by making use of a weighted combination of local pixel values of the same color, along the identified direction.

The patent [24] is composed by an interpolation step followed by a correction step. The authors consider the luminance channel as proxy for green color, and the chrominance channel as proxy for red and blue. Since the luminance channel is more accurate, it is interpolated before the chrominance channels. The luminance is interpolated as accurate as possible in order to not produce wrong modifications in the chrominance channels. However, after the interpolation step, luminance and chrominances are orderly refined. The interpolation phase is based on the analysis of the gradients in four directions (east, west, north and south), involving both luminance and chrominance channels, as it is apparent from the (17).

$$\begin{aligned}\Delta W &= |2L_{(x-1,y)} - L_{(x-3,y)} - L_{(x+1,y)}| + |C_{(x,y)} - C_{(x-2,y)}| \\ \Delta E &= |2L_{(x+1,y)} - L_{(x-1,y)} - L_{(x+3,y)}| + |C_{(x,y)} - C_{(x+2,y)}| \\ \Delta N &= |2L_{(x,y-1)} - L_{(x,y-3)} - L_{(x,y+1)}| + |C_{(x,y)} - C_{(x,y-2)}| \\ \Delta S &= |2L_{(x,y+1)} - L_{(x,y-1)} - L_{(x,y+3)}| + |C_{(x,y)} - C_{(x,y+2)}|\end{aligned}\quad (17)$$

Since the aim is to interpolate along edges and not across them, an inverted gradient function $fgrad=(1/gradient)$ is formed, which allows to weight more the smallest gradients, in order to follow the edge orientation. The interpolation of missing luminance values is performed using the normalized inverted gradient functions which weight both luminance and chrominance values in the neighborhood. It is important to note that chrominance values are used in the interpolation of luminance to get a more accurate estimation. Similarly, chrominances are interpolated by using both luminance and chrominance data. The correction step comprises the luminance correction first, and then the chrominance correction.

The patent [25] stresses the importance of preserving the sharpness at edge locations. In fact, edge detection is used during the interpolation in order to reconstruct the missing color channels differently depending on edge direction and considering the edgeness of the pixel. Since the edge orientation is estimated on the green channel, a first simple interpolation of the green missing values is achieved by using a median operator.

With reference to Fig. (14), dashed lines 2, 5, 8 and 11 are the four directions for which edge direction calculations are achieved (horizontal, vertical, positive diagonal and negative diagonal). The remaining dashed lines can be used in calculating the four edge direction values.

Before performing any interpolation, the method determines if the processed pixel belongs to an edge. It

calculates a set of differences between pairs of the same color on the Bayer pattern, and then evaluates if one of them exceeds a predetermined edge threshold. If no edge is detected, the interpolation is performed through a simple average of the surrounding pixels. On the contrary, if the processed pixel is marked as belonging to an edge, the system reconstructs the missing color values as an average of the surrounding correspondent colors plus a correction factor based on the directive Laplacian applied to the original color channel:

$$New_value = LPF_neighbor + Edge_HPF_Original_color \quad (18)$$

It is important to specify that the first term ($LPF_neighbor$) does not depend on the identified edge direction, because it is a simple average of the surrounding values.

The directional contribution is given only by the second term of the equation (18), which changes according to the edge direction.

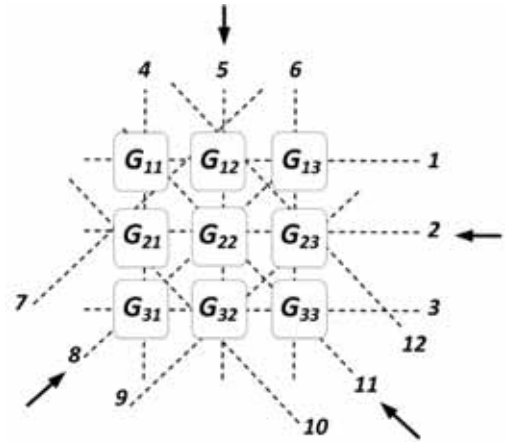


Fig. (14). Example of 3x3 kernel.

Moreover, in contrast with the most of prior art systems, the sensed pixel value may be replaced by an interpolated value, thus eliminating the imbalance resulting from the use of sensed values for one of the colors and interpolated values of the remaining two colors.

The [26], similarly to the previous one, has the aim of producing images with sharp edges. And also in this case a high frequency component, derived from the sensed color channel, is added to the low frequency component of the interpolated channels. However, the differences between these two techniques are many.

This patent takes into account 8 different directions, as it shown in Fig. (15), and uses 5x5 elliptical Gaussian filters to interpolate the low frequency component of each color channel (even the sensed one). For each available direction there is a different Gaussian filter, having the greater coefficients along the identified direction. These filters have the advantage of interpolating the missing information without generating annoying jaggy edges.

After having computed the low frequency component, for each color channel, an enhancement of the high frequencies content is obtained taking in account the spectral correlation

(eq.3). In particular, a correction term is calculated as the difference between the original sensed value and its low pass component, which is retrieved through the directional Gaussian interpolation. In case of a G central pixel, the correction term is calculated through the following formula:

$$\Delta_{Peak} = G - G_{LPF} \quad (19)$$

This correction term is then added to the low frequency component of the channels to be estimated:

$$H = H_{LPF} + \Delta_{Peak} \quad (20)$$

where H indicates both R and B channels.

It is straightforward to understand that the low frequency component, in this patent, is calculated according to the identified direction, so it is less affected by false colors than previous inventions. Moreover, this solution provides a simple and effective method for calculating direction and amplitude values of spatial gradients, without making use of a first rough interpolation of the G channel. More specifically, 3×3 Sobel operators are applied directly on the Bayer pattern to calculate horizontal and vertical gradients. The orientation of the spatial gradient at each pixel location is given by the following equation:

$$or(x,y) = \begin{cases} \arctan\left(\frac{P^*Sobel_y(x,y)}{P^*Sobel_x(x,y)}\right) & \text{if } P^*Sobel_x(x,y) \neq 0 \\ \frac{\pi}{2} & \text{otherwise} \end{cases} \quad (21)$$

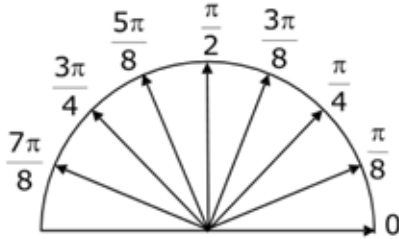


Fig. (15). Quantized directions for spatial gradients.

where P^*Sobel_y and P^*Sobel_x are the vertical and horizontal Sobel filtered values, at the same pixel location. The orientation $or(x,y)$ is quantized in 8 predefined directions (see Fig. (15)). Since the image could be deteriorated by noise, and the calculation of direction could be sensitive to it, a more robust estimation of direction is needed. For this reason, Sobel filters are applied on each 3×3 mask within a 5×5 window, thus retrieving nine gradient data. In addition to the orientation, the amplitude of each spatial gradient is calculated, by using the following equation:

$$mag(x,y) = \left(P^*Sobel_x(x,y)\right)^2 + \left(P^*Sobel_y(x,y)\right)^2 \quad (22)$$

The direction of the central pixel is finally derived through the “weighted-mode” operator, which provides an estimation of the predominant amplitude of the spatial gradient around the central pixel. This operator substantially reduces the effect of noise in estimating the direction to use in the interpolation phase. It is important to stress that this

invention achieves the interpolation of the three color channels in a single step, without requiring a first interpolation of the G channel.

Pattern Based

Another class of demosaicing methods includes the pattern based interpolation techniques. Such algorithms perform, generally, a statistical analysis, by collecting actual comparisons of image samples with the corresponding full-color images. Chen *et al.* ([27]) propose a method to improve the sharpness and reduce the color fringes with a limited hardware cost. The approach consists of two main steps:

1. Data training phase:

- a. Collecting samples and corresponding full-color images;
- b. Forming pattern indexes, by selecting the concentrative window for each color in the Bayer samples and quantizing all the values on the window;
- c. Calculating the errors between the reconstructed pixels and the actual color values;
- d. Estimating the optimal combination of pattern indexes to be sorted into a database.

2. Data practice phase:

For each pixel a concentrative window is chosen, and within it, the pixels are quantized in two levels (Low, High) to form a pattern index, as shown in Fig. (16). This index is then used as key for the database matching.

During the data training phase, the proposed method assumes that the reconstructed value is function of the original value and the feasible coefficient set, which can be expressed as:

$$Rec_{value} = Orig_{value} * feasible_coefficient_set / (sum_of_coefficients) \quad (23)$$

Once the value has been calculated for each feasible coefficient set, the system chooses the set having the minimal error between the calculated values and the real value. These results are then stored into the database. During the data-practice phase, the reconstruction is based on color differences rules applied to the pixel neighborhood. The results of the proposed method highlight a similar behavior to the ones proposed by Gunturk [8] and Lu [28], although the proposed method has less complexity.

A simpler technique [29] uses a plurality of stored interpolation patterns. To select the correct interpolation pattern, an analysis of the input signal is performed using gradient and uniformity estimation.

In practice, by first the G channel is interpolated using the 8 stored fixed patterns (Horizontal, Vertical, the two Diagonals and the four corners). To achieve this purpose the uniformity and the gradient are estimated in the surrounding of the selected G pixel. The minimum directional data estimation $G_v(i)$ ($i \in [0,8]$), obtained through the eight fixed patterns, defines the best match with the effective direction. For example, Fig. (17) shows an interpolation pattern in which low luminance pixels are arranged along the diagonal.

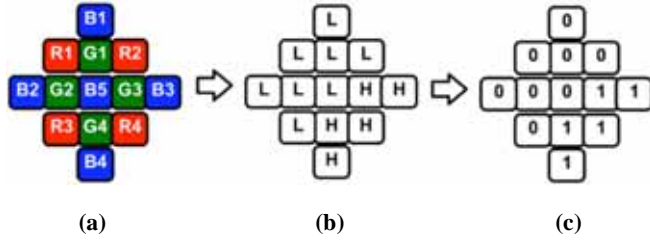


Fig. (16). Relationship between a color filter array and a concentrative window. (a) Bayer Pattern; (b) Quantization of acquired samples in two levels: Low (L) and High (H); (c) Resulting pattern index.

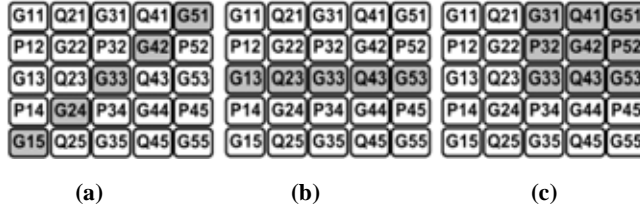


Fig. (17). Some samples of interpolation patterns.

The directional data $G_v(I)$, which represents a numerical value of the similarity between the surround of the pixel to be interpolated and the interpolation pattern, is obtained through the following expression:

$$G_v(I) = (|G_{33} - G_{51}| + |G_{33} - G_{42}| + |G_{33} - G_{24}| + |G_{33} - G_{15}|) / 4. \quad (24)$$

The remaining seven directional data are calculated in a similar manner, taking into account the fixed direction. The smallest directional data from $G_v(1)$ to $G_v(8)$ identifies the interpolation pattern which is the best fit to the image neighborhood of the pixel to be interpolated.

When only one interpolation pattern providing the smallest directional value is present, it is chosen to perform the interpolation. On the contrary, when two or more interpolation patterns provide the smallest directional value, a correlation with the interpolation patterns of the surrounding pixels, whose optimum interpolation pattern has already been determined, is taken into consideration.

Specifically, if one of the interpolation patterns having the smallest value is the interpolation pattern of one surrounding G pixel, this pattern is chosen for performing the interpolation. Otherwise it is impossible to determine a specific pattern to use for the interpolation, and thus a simple isotropic low pass filter is applied.

If $G_v(I)$ is the smallest directional value, the interpolation is achieved through the equation:

$$\begin{aligned} G_0 &= (G_{15} + G_{24} \times 2 + G_{33} \times 2 + G_{42} \times 2 + G_{51}) / 8 \\ H_0 &= (H_{14} + H_{34} + H_{32} + H_{52}) / 4 \\ J_0 &= (J_{23} + J_{25} + J_{43} + J_{41}) / 4 \end{aligned} \quad (25)$$

where H and J represent the R and B or B and R values.

If it is impossible to determine a specific pattern, the interpolation is performed by the following formula:

$$\begin{aligned} G_0 &= (G_{22} + G_{24} + G_{42} + G_{44} + G_{33} \times 4) / 8 \\ H_0 &= (H_{34} + H_{32}) / 2 \\ J_0 &= (J_{23} + J_{43}) / 2 \end{aligned} \quad (26)$$

Once the missing values for the G pixels have been processed, the algorithm calculates the missing values for the R and B pixels. If the interpolation patterns, estimated for the G pixels already processed, in the surrounding of the R or B pixel describe a fixed direction (several patterns indicate the same direction) then this pattern is used to perform the interpolation. Otherwise the numerical directional data are estimated. Like the G case, eight different interpolation patterns are stored in the interpolation storage memory and a directional data value is computed for each of these patterns. When there are two or more patterns having the smallest directional data value, correlations with the interpolation patterns of the already interpolated G pixels are evaluated. The reason why G pixels are taken into consideration instead of R and B pixels is that G pixels are more suitable for pattern detection than R and B pixels.

To complete the process with an enhancement of the final quality, after interpolation, edge enhancing and other signal processing can be performed by processing signals depending on respective interpolation patterns, because these are stored into the memory.

This technique is very robust to noise, because it takes into consideration the interpolation patterns of the already processed pixels, but introduces jagged edges in abrupt diagonal transitions, due to the equations used in the interpolation step.

Iterative

In this category we collect all approaches that derive interpolation through an iterative process able to find after a limited number of cycles the final demosaiced image. In particular, we mention the US patent [30-32] co-authored by the same list of authors, where, starting from an initial rough estimate of the interpolation, the input data are properly filtered (usually by employing a combination of directional high-pass filters with some global smoothing) to converge versus stable conditions. The three methods proceed in different ways with respect to the local image analysis but share the overall basis methodology.

In [30] a color vector image is formed containing the original Bayer values. After an initial estimate of the real RGB original value of each pixel such quantity is updated by taking into account two different functions: "roughness" and "preferred direction". The final missing color are defined by finding the values that minimize a weighted sum of *Rough* and *CCF* (Color Compatibility Function) functions over the image by using the following formula:

$$Q = \sum_{m,n} \text{Rough}(m,n) + \lambda \sum_{m,n} \text{CCF}(m,n) \quad (27)$$

where λ is a positive constant while *Rough* (m,n) is defined in this case as the local summation of approximated local

gradients and $CCF(m,n)$ is a function that penalizes local abruptly changes. By using the classic Gauss-Siedel approach the method converge after 4-5 iterations.

In [31], [32] the luminance channel is properly extracted from input Bayer data and analyzed in a multiscale framework by applying smoothing filtering along preferred directions. Chrominance components are smoothed by isotropically smoothing filters. The final interpolated image is obtained after a few iterations. Just before to start a new iteration the pixel values are reset to the original (measured) values.

Some proposals infer locally the interpolation parameters by making use of regression analysis previously performed on different regions of the same image [33-35]. The technique described in [33] makes use of a linear prediction from the raw color sensor value at the current pixel location. The Bayer image is divided in several regions. For each region the missing values are calculated by interpolation or by regression (making use of partial data previously obtained on other regions). Differently than [33] where local regression parameters are statistically computed in a robust way, in [34] the local analysis is based only on simple local energy values (variance and/or gradient).

Other Approaches

Finally a list of methods that propose some demosaicing approaches making use of alternative strategies is presented. The method proposed in [36], transforms input RGB data in La^*b^* color space just to work in a perceptual uniform way. The interpolation is achieved by means of wavelet decomposition. Of course, the additional effort needed to transform the input data into another domain, limits the implementation of such techniques in real environment. In this context, we mention also some academic solutions [37,38], that works in the Fourier domain.

In [39] the interpolation process is achieved by considering the results of a bilateral filtering [40] together with classic convolution kernels able to preserve high frequency details without introducing color artifacts. The bilateral filtering is able to smooth data preserving edge details according to local consideration by means of a non-linear process.

Another recent technique [41] derives final RGB data by pre-processing the input data by DDT (Data Dependent Triangulation). Such technique has been successfully applied both for image interpolation [42] and for raster to vector conversion [43].

Different CFA patterns are considered in [44] where successive raw of Bayer data are slightly shifted along horizontal directions while in [45] a color filter able to acquire the infrared component of light is used just to better approximate local luminance components. Finally, in technique in [46] the interpolation process is coupled with some heuristics able to compensate the chromatic aberration of system.

ANTI_ALIASING

Due to constraints on cost-effective solutions, color interpolation methods usually result in color edge artifacts in the image.

There are numerous ways in the prior art for reducing color artifacts in digital images. There are also numerous patents that describe color artifacts (or moiré pattern) reduction methods using optical blur filters in digital cameras to avoid aliasing in the first place. However, these blur filters also eliminate spatial details in the image that may not be recovered by further image processing algorithms.

For obtaining better performances, as previously mentioned the antialiasing step often follows the color interpolation process, as a postprocessing step (see Fig. (18)) in the RGB domain.

In [47], the authors propose a method that exploits the original uncorrupted Bayer CFA data present in the demosaiced image with a localized color ratio model to correct erroneous color components produced by CFA interpolation.

In [48] the authors propose a method to eliminate the false color and zipper effect based on an adaptive scheme allowing to determine the specific artifact affecting the pixels. The authors use the spectral correlation between color planes to detect and reduce the artifacts. The block diagram representing the demosaicing artifact removal algorithm is shown in Fig. (19). Before processing each pixel, the zipper detector block produces a control signal which enables either the false colors removal algorithm or the zipper effect removal algorithm.

Kakarala *et al.* in [49] propose a color aliasing artifact reduction (CAAR) algorithm. The CAAR algorithm computes the L level wavelet transform for the demosaiced color planes R , G and B . Then, the CAAR algorithm estimates the correct color value at each pixel location for the colors not associated with that pixel location. This process is repeated for each of the colors. In addition, the CAAR algorithm performs an inverse wavelet transform on each of the color planes themselves, so that the pixel values of the color associated with each pixel location are not altered.

Another class of patents works on the $YC_R C_B$ domain which is less correlated. While edges still tend to be strong in the Y (luminance) component, they tend to be weak in the C_R and C_B (chrominance) components. In general, C_R and C_B planes are smoother than the RGB plane, thus they should be more suitable for removing false colors.

Some patents propose a blurring on the chrominance planes to obtain a cost effective solution to aliasing, while the luminance remains unchanged.

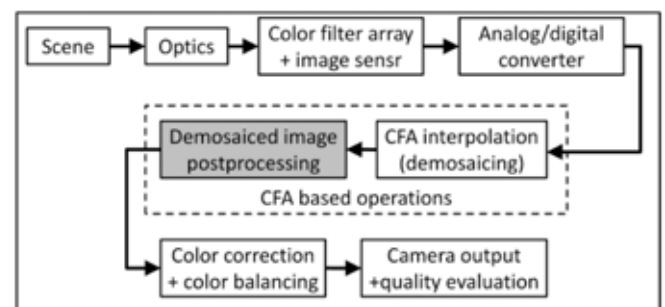


Fig. (18). IGP with Antialiasing postprocessing.

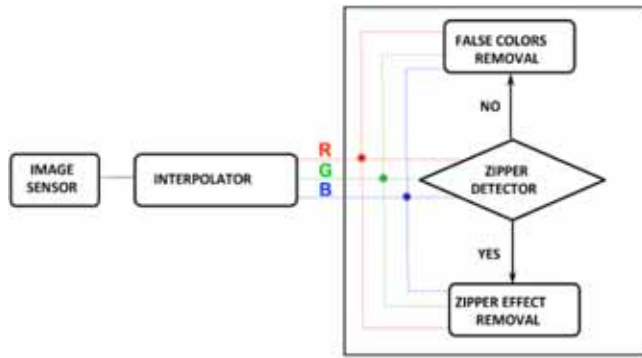


Fig. (19). Block diagram of the anti-aliasing algorithm proposed in [48].

In [50] a method of reducing color aliasing artifacts using luminance and chrominance signals to separate the image into textured and nontextured regions is proposed. It is based on the following steps:

- downsampling the chrominance signals and texture region mapping;
- producing cleaned chrominance signals in the textured regions;
- producing cleaned chrominance signals in the nontextured regions;
- upsampling the noise cleaned chrominance signals;
- using the luminance and upsampled noise-cleaned chrominance signals to provide a color digital image having reduced color aliasing artifacts.

QUALITY EVALUATION

According to the patenting purposes, the intrinsic quality of a patented method is not a differentiating factor. Anyway, the quality evaluation is of course an important aspect to assess the effectiveness of the proposed solutions.

The problem of quantitatively evaluate the quality of demosaicing methods is related to the test set to be used and the methodology to adopt. As well known, the quality evaluation is always an open task [51] since the start of photographic era. The usage of no reference metrics to assess the quality is complex, because we could only define a priori some features to be measured (blurriness, zipper, false colors). Some papers try to define some no reference metrics for the demosaicing [52,53]. These approaches, better follows the characteristics of the Human visual system, but at the same time make more difficult to compare already existing solutions. The easiest solution is to use reference metrics, as shown in Fig. (20). The original image is synthetically bayerized, and then it is interpolated through the algorithm which has to be evaluated. The demosaicing result is then compared to the corresponding reference image, thus computing an error image. The more the result is similar to the reference image, the more the algorithm is assumed well performing.

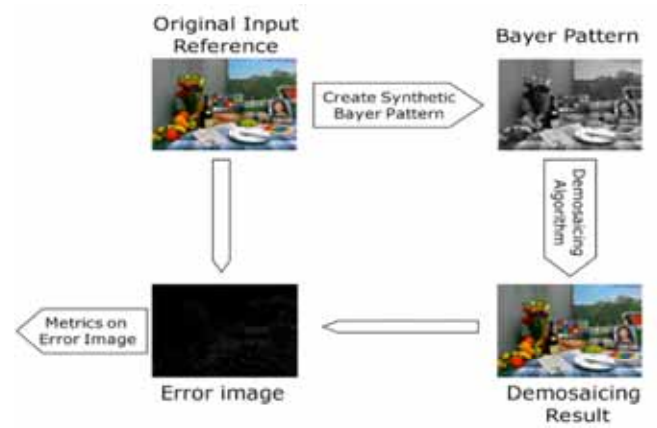


Fig. (20). Reference Metric Model.

Nowadays all the published papers on this topic use the same image test set [54]. These images contains a lot of edges and textures, thus they are useful for highlight how the various algorithms handle the edges. The Kodak image test set is full color, thus the CFA images are simulated by subsampling the color channels according to the Bayer Pattern.

Another test image used to evaluate both directivity and high frequencies behavior is the Circular Plate Zone (CPZ) defined as follows:

$$f(x, y) = C_1 \cdot \cos\left(\frac{\pi}{N^2} (x^2 + y^2) \cdot f_{\max}\right) + C_2 \quad (28)$$

where $f_{\max} = N$

and C_1, C_2 are constant

which corresponds to the image shown in Fig. (21a). Near the Nyquist frequencies it is possible to evaluate how various methods reconstruct details. The image is grey scale, thus each artifact appears as false color, as shown in Fig. (21b) and Fig. (21c).

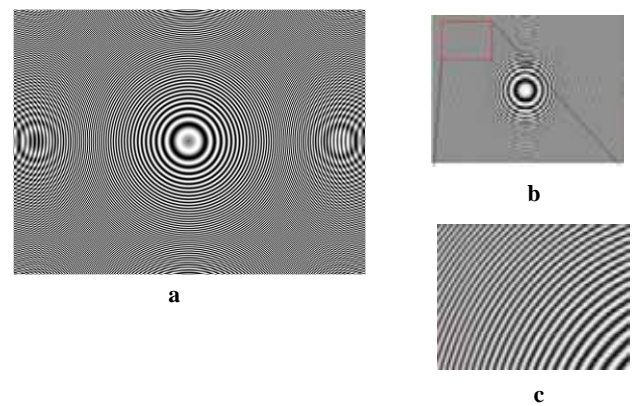


Fig. (21). Circular Plate Zone (CPZ).

Once RGB images are reconstructed a lot of similarity metrics can be applied.

The most used objective metric is the PSNR (Peak signal to Noise Ratio). Given two $N \times M$ images A (reference) and B (interpolated), the PSNR is expressed as:

$$PSNR = 20 \times \log_{10} \left(\frac{Max_Value}{\sqrt{\frac{1}{c \times N \times M} \sum_{s=1}^{N \times M} (A_{s,i} - B_{s,i})^2}} \right) \quad (29)$$

where s represent the index of the color channel (in this case R , G , and B) and c is the number of channels ($c=1$ for grayscale images, $c=3$ for RGB images).

Higher values (expressed in decibel) of the PSNR generally imply better quality.

The PSNR can be computed for each color channel [2], [55] or globally ([37,56]).

Since the main problems of the demosaicing approaches occur across edges or textured zones, other papers suggest to firstly evaluating the edge zones, and then to separately compute the PSNR in edge and flat zones [57]. Of course in this case almost all methods have a high and similar PSNR value in flat zones, while the differentiating value is the PSNR in edge zones.

Another measure is the *CIELab* ΔE_{ab}^* which measures the Euclidean distance between the original image and the interpolated one in the *CIELab* space [58]. L , a and b are transformation of X, Y and Z tristimulus values, which contain information about the absorption level of cone receptors. For this reason this space better follows the human retina perceptual difference sensitivity. The *CIELab* ΔE_{ab}^* is defined as:

$$\Delta E_{ab}^* = \frac{1}{N \times M} \sum_{i=1}^{N \times M} \|A_i^{Lab} - B_i^{Lab}\| \quad (30)$$

Where A_i^{Lab} and B_i^{Lab} are pixels expressed in the *CIELab* color space. Values of ΔE_{ab}^* greater than 2.3 indicate that the differences are visible, mean differences greater than 10 imply that differences between the two images are so high that the comparison is not worthwhile.

Nevertheless the above mentioned metrics, applied to synthetic subsampled scenes, a real developed system should consider that many other factors affect the final quality, such as the optics, the noise, the white balance, etc.

CURRENT & FUTURE DEVELOPMENTS

The demosaicing step in a single-sensor imaging pipeline is, of course, a key aspect of the overall design of any imaging device. In this paper we have briefly summarized the main peculiarities of some recent patents in the field. The demosaicing solutions have been grouped according to their intrinsic processing strategies as method respectively based on interchannel correlation, edge, pattern and iterative. Also we provided a brief review about antialiasing strategies together with some comments about quality evaluation. The demosaicing patented solutions, differently than recent academic techniques, have to be fast and simple to be implemented inside a system with low capabilities (e.g.,

memory, CPU, low-power, ...). It should be noted, however, that some patents contains some techniques that were at some point academic. Also the increased advances in terms of speed and computational power of recent imaging devices allows to easily predict that in the future some new "heavy" techniques ([7]) could be also implemented without any restrictions. In the following we list some interesting issues that could be addressed to design the subsequent generation of demosaicing approaches ([59]).

First of all it would be interesting to implement a full and extensive comparison between the intrinsic capabilities of an "ideal" color interpolation strategy (where no constraints are considered) and recent papers and patents on demosaicing. We claim that a real advance in the field of demosaicing can be reached just considering novel and powerful algorithms that works considering also the real semantic content of the scene, adapting the interpolation strategy both to local patterns and global features of the scene under acquisition [60]. Also the integration of the demosaicing step with other image processing solutions inside the Image Generation Pipeline (e.g., white balancing, denoising, matrixing, ...) could be exploited ([61]) and coupled with some computational methods able to detect and measure (correcting it) the various artifacts due to a coarse interpolation strategy.

REFERENCES

- [1] Bayer, B.E.: US3971065 (1976).
- [2] Lian N-X, Chang L, Tan T-P, Zagorodnov Z. Adaptive filtering for color filter array demosaicking. *IEEE Trans On Image Proc* 2007; 16(10): 2515-2525.
- [3] Sakamoto T, Nakanishi C, Hase T. Software pixel interpolation for digital still cameras suitable for a 32-bit MCU. *IEEE Trans on Consumer Electronics* 1998; 44(4): 1342-1352.
- [4] Gunturk BK, Glotzbach J, Altunbasak Y, Schafer RW, Mersereau RM. Demosaicking: Color Filter Array Interpolation. *IEEE Signal Proc Magazine* January 2005; 22(1): 44-54.
- [5] Yuk CKM., Au OC., Li RYM., Lam S-Y. Color demosaicking using direction similarity incolor difference spaces. *In Proc IEEE International Symposium on Circuits and Systems* 2007; 1281-1284.
- [6] Lian N., Chang L., Tan YP. Improved color filter array demosaicking by accurate luminance estimation. *IEEE Proc Int Conf Image Proc* 2005; 41-44.
- [7] Menon D., Calvagno G. Demosaicking based on wavelet analysis of the luminance component. *In Proc IEEE International Conf on Image Proc*, September 2007; 181-184.
- [8] Gunturk BK, Altunbasak Y, Mersereau R. Color plane interpolation using alternating projections. *IEEE Trans on Image Proc* September 2002; 11(9): 997-1013.
- [9] Alleysson D, Susstrunk S, Herault J. Linear demosaicking inspired by human visual system. *IEEE Trans on Image Proc* April 2005; 14(4): 439-449.
- [10] Chaix de Lavarène B., Alleysson D., Durette B., Héroult J. Efficient demosaicking through recursive filtering. *In Proc IEEE International Conf on Image Proc*, September 2007; 189-192.
- [11] Nomura Y., Nayar SK. A VQ-Based Demosaicking by Self-Similarity. *IEEE International Conf on Image Proc*, September 2007, 457-460.
- [12] Hunter, A.A., Pollard, S.B.: US20030086606A1 (2003).
- [13] Juenger, A.K., Metz, W.: US005778106A (1998).
- [14] Lee, H.: US20040196395A1 (2004).
- [15] Chen, T.: A study of spatial color interpolation for digital still cameras. <http://scien.stanford.edu/class/psych221/projects/99/tingchen/>.
- [16] Nohda, S.: US20016295087B1 (2001).
- [17] Chen, Z.: US20026487309B1 (2002).
- [18] Hamilton, J.F. Jr., Adams, J.E. Jr.: US05629734A (1997).
- [19] Kalevo, O., Rantanen, H.: US20040218073A1 (2004).
- [20] Nguyen, T.: US20020167602A1 (2002).

- [21] Chen, Z.: US2003657061B1 (2003).
- [22] Tajima, K.: US20050058361A1 (2005).
- [23] Utagawa, K.: US20036563538-B1 (2003).
- [24] Nilsson, A. Nordblom, P.: US20070292022A1 (2007).
- [25] Subbotin, I.: US20070153106A1 (2007).
- [26] Messina, G., Guarnera, M.I., Tomaselli, V., Bruna, A.R., Spampinato, G., Castorina, A.: US20077305123-B2 (2007).
- [27] Chen, C., Lin, C., Wu, C., Yin, Z., Wang, C.: US20077292725B2 (2007).
- [28] Lu W, Tan Y. Color filter array demosaicking: New method and performance measures. *IEEE Trans on Image Proc* 2003; 12(10): 1194-1210;
- [29] Kimura, K., Yoshida, T.: US20050007470A1 (2005).
- [30] Keren, D.: US20036625305B1 (2003).
- [31] Hel-or, Y., Keren, D.: US20026404918B1 (2002).
- [32] Hel-or, Y., Keren, D.: US20036618503B2 (2003).
- [33] Zhang, X., Kakarala, R., Sachs, T.S.: US20067079705B2 (2006).
- [34] Baharav, I., Kakarala, R., Vook, D.W.: US20040201721A1 (2004).
- [35] Zhang, X., Kakarala, R.: US20040086177A1 (2004).
- [36] Tao, B.: US20046809765B1 (2004).
- [37] Menon D, Andriani S, Calvagno G. Demosaicing with directional filtering and a posteriori decision. *IEEE Trans Image Proc* 2007; 16(1): 132-141.
- [38] Lavarene BC., Alleysson D., Durette B., Herault J. Demosaicing based on spectra consideration: Efficient demosaicing through recursive filtering. *In Proceedings of IEEE ICIP07* 2007; 189-192.
- [39] Keshet, R., Barash, D., Shaked, D., Elad, M., Kimmel, R.: US20046816197B2 (2004).
- [40] Tomasi C., Manduchi R. Bilateral filtering for gray and color images. *Proceedings of the 1998 IEEE international conference on computer vision*, Bombay, India, 1998.
- [41] Spampinato, G., Messina, G., Bruna, A.R., Guarnera, M.: US20077305148B2 (2007).
- [42] Su D, Willis P. Image interpolation by pixel-level data-dependent triangulation. *Computer Graphics Forum*, 23 February 2004.
- [43] Battiato S., Gallo G., Messina G., SVG rendering of real images using data dependent triangulation. *In Proceedings of Spring Conference on Computer Graphics* 2004; 191-198.
- [44] Bishay, M.M., Chung, R.M.: WO9946730A1 (1999).
- [45] Ohki, M., Mitsunaga, T., Yoshimura, S., Kasai, M.: US20070153099A1 (2007).
- [46] Stavely, D.J., Whitman, C.A., Sobol, R.E., Matherson, K.J.: US20040240726A1 (2004).
- [47] Lukac R, Martin K, Plataniotis KN. Demosaicked image postprocessing using local color ratios. *IEEE Trans On Circuits Syst Video Technol* 2004; 14(6): 914- 920.
- [48] Guarnera, M., Messina, G., Tomaselli, V.: US20060087567A1 (2006).
- [49] Kakarala, R., Baharav I.: US20046819801B2 (2004).
- [50] Adam, J.E. Jr., Hamilton, J.F., Smith, C.M.: US2004046894A1 (2004).
- [51] Engeldrum PG. Image quality modeling: Where are we? *IS&T's 1999 PICS Conference* 1999.
- [52] Guarneri I, Guarnera MI, Bosco A, Santoro G. A perceptual quality metric for color interpolated images. *SPIE Proceedings of Electronic Imaging 05*, San Jose, California, USA, January 2005; 16-20.
- [53] Longre P., Zhang X., Delahunt PB., Brainard D.H. Perceptual assessment of demosaicing algorithm performance. *Proc of IEEE* 2002; 90: 123-132.
- [54] Kodak Images Database, <http://r0k.us/graphics/kodak/>;
- [55] Pei S-C, Tam I-K. Effective color interpolation in CCD color filter arrays using signal correlation. *IEEE Trans On Cir Sys Video Tech* 2003; 13(6).
- [56] Muresan DD, Parks TW. Demosaicing using optimal recovery. *IEEE Trans on Image Proc* 2005; 14(2).
- [57] Mukhejee J., Moore MS., Mitra SK. Color demosaicing with constrained buffering. *Int Symp on Signal Processing and its Applications (ISSPA)* Kuala Lumpur, Malaysia, August, 2001.
- [58] Recommendation on Uniform Color Spaces, Color Difference Equations, Psychometric Color Terms." C.I.E., Supplement n.2 to CIE publication n.15 (E-131)1971/(TC-1.3), 1978.
- [59] Lukac R. Single-sensor imaging: Methods and applications for digital cameras - CRC Press - 2008.
- [60] Battiato S., Farinella GM., Gallo G., Ravi D. Scene categorization using bags of textons on spatial hierarchy. *IEEE International Conference on Image Processing*, October 2008.
- [61] Hirakawa K., Parks TW. Joint demosaicing and denoising. *In Proc International Conf on Image Proc*, September 2005; 309-312.