Data Adaptive Filters for Demosaicking: A Framework

Rastislav Lukac, *Member, IEEE*, and Konstantinos N. Plataniotis, *Senior Member, IEEE*

Abstract — *A new demosaicking framework for single-sensor imaging devices operating on a Bayer color filter array (CFA) is introduced and analyzed. An efficient data adaptive filtering concept in conjunction with the refined spectral models constitute the base for the proposed framework. Using a different form of the function mapping the aggregated absolute differences among the CFA inputs to the edge-sensing weighting coefficients, the framework allows to design fully automated demosaicking solutions suitable for common digital imaging apparatus, and alternatively, the proposed solutions can also be used to support PC-based demosaicking of the raw CFA images. Thus, the framework can be seen as an universal tool satisfying the needs of the end-users for i) the instant access and visualization of the captured images, and ii) the interactive processing of the raw sensor data. Moreover, the proposed framework is relatively easy to implement in either software or hardware. Experimental results indicate that the proposed framework exhibits excellent performance in terms of the commonly used objective criteria and at the same time it produces demosaicked images with impressive visual quality. [1]*

Index Terms — **Single-sensor imaging, digital camera, camera image processing, Bayer pattern, color filter array interpolation, demosaicking, data adaptive filters.**

I. INTRODUCTION

In the last few years, digital imaging devices have become popular over the traditional film cameras and been widely embedded in consumer electronics ranging from the conventional digital cameras, to pocket devices, to mobile phones, and imaging devices for automotive and surveillance applications. It is therefore not a surprise that digital capturing capabilities are required today also in a wide range of applications, such as computer vision, medical imaging, astronomy, etc.

To capture the image scene, digital cameras use image sensors, usually charge coupled devices (CCD) or complementary metal oxide semiconductor (CMOS) sensors. Following the trichromatic theory of color vision, an arbitrary color is matched by superimposing appropriate amounts of three-primary colors [1],[2]. Since sensor is a monochromatic device, professional digital cameras acquire color information using three sensors (Fig. 1) with Red (R), Green (G) and Blue (B) color filters having different spectral transmittances [3].

The authors are with The Edward S. Rogers Sr. Department of ECE, University of Toronto, Canada.

Corresponding Author: Dr. Rastislav Lukac, Bell Canada Multimedia Laboratory, Room BA 4157, The Edward S. Rogers Sr. Department of ECE, University of Toronto, 10 King's College Road, Toronto, Ontario, M5S 3G4, Canada (e-mail: lukacr@ieee.org)
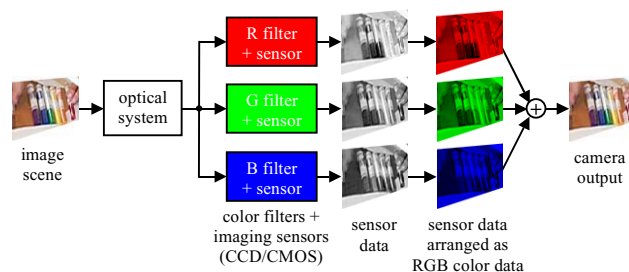
Fig. 1. Three-sensor digital camera architecture.
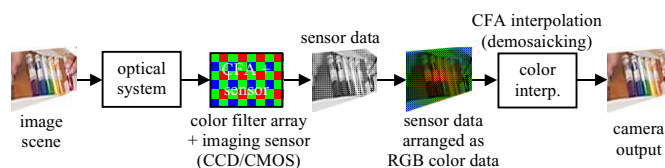

Fig. 2. Single-sensor digital camera architecture.

It is known that the sensor is usually the most expensive component of the digital camera taking from 10% to 25% of the total cost of the device. Therefore, digital camera manufacturers reduce cost and complexity using a single CCD or CMOS sensor with a color filter array (CFA). Using this hardware solution (Fig. 2), each pixel of the raw, CFA image has its own spectrally selective filter. The specific arrangements of color filters in the CFA [4],[5] vary between the camera manufacturers which use not only RGB CFAs, however, the patterns with complementary Cyan (C), Magenta (M), Yellow (Y) colors, or four-color CFAs formed through mixed primary (RGB) and complementary (CMY) colors or the color primaries and the fourth – spectrally shifted color are in the use today as well. Among these, the Bayer pattern (Fig. 3) [6] is commonly used due to simplicity of the subsequent processing steps. This pattern contains twice as many G components compared to R or B components reflecting the fact that the spectral response of G filters is close to the luminance response of human visual system [7],[8],[9]. Since each color pixel of the RGB Bayer CFA image contains only a single measurement, the two missing color components must be estimated from the adjacent pixels. This process is called CFA interpolation [9],[10], or demosaicking [11],[12], and is an integral element in single-sensor imaging. Depending on the demosaicking algorithm employed, the cost of the device as well as the quality of the output can vary significantly.

To develop the cost-effective and powerful approach for color interpolation in digital still cameras, the data-adaptive demosaicking framework is introduced. The structural content of the CFA image is tracked based on the aggregation concept defined over the absolute differences between the CFA inputs
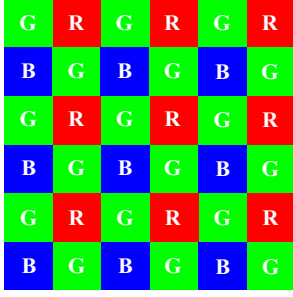
Fig. 3. Bayer CFA pattern with the GRGR phase in the first row.

(a)   (b)

Fig. 4. Aggregation concept defined here over the four-neighborhoods: (a) rectangular lattice, (b) quincunx lattice.

[13]. The aggregated statistics are further mapped into the weighting coefficients used to direct the interpolation process. To avoid color shifts, the spectral characteristics of the image input are utilized during processing through the use of the spectral model. Based on the actual form of the mapping and spectral model employed, thy numerous solutions differentiating in terms of functionality as well as performance are designed within the proposed framework.

The rest of this paper is organized as follows. In Section II, data-adaptive filtering concept used in numerous digital image processing tasks is described. The proposed demosaicking framework is introduced in Section III. Motivation and design characteristics are discussed in detail. In Section IV, the Bayer pattern demosaicking solutions designed within the proposed framework are tested using a variety of color images. Finally, conclusions are drawn in Section V.

II. DATA-ADAPTIVE FILTERING CONCEPT

Let us consider the population of the RGB vectors $\mathbf{x}_{(i,j)} = [x_{(i,j)1}, x_{(i,j)2}, x_{(i,j)3}]$ with $x_{(i,j)k}$ indicating the R $(k=1)$, G $(k=2)$ and B $(k=3)$ component. Using the data-adaptive vector filtering concept [2],[14],[15], the set of $\mathbf{x}_{(i,j)}$, for $(i,j) \in \varsigma$ denoting the spatial location inside the filtering window centered in (r,s), is used to estimate the sample $\mathbf{x}_{(r,s)}$ as follows:

$$\mathbf{x}_{(r,s)} = f\left(\sum_{(i,j)\in\varsigma} w'_{(i,j)}\mathbf{x}_{(i,j)}\right) \qquad (1)$$

where $f(.)$ is a nonlinear function that operates over the weighted average of the input set. To ensure that the filter output is an unbiased estimator and produces the samples within the desired intensity range, the weights $w_{(i,j)}$ associated with the vectorial inputs $\mathbf{x}_{(i,j)}$ are normalized using

$$w'_{(i,j)} = \frac{w_{(i,j)}}{\sum_{(g,h)\in\varsigma} w_{(g,h)}} \qquad (2)$$

In this design the weights provide the degree to which an input vector contributes to the output of the filter [2],[14]. The relationship between the sample under consideration and each
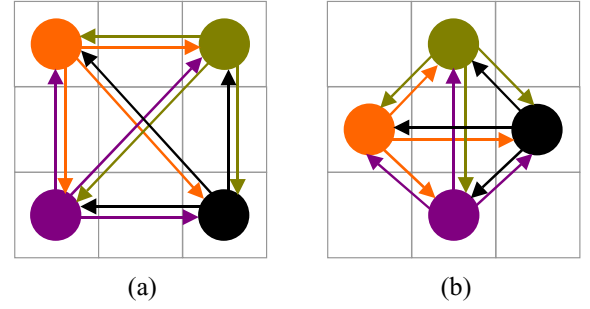
sample positioned at the window locations ς is reflected in the decision for the weighting coefficients $w_{(i,j)}$. The weights of the filter are adaptively determined using functions of a distance (or similarity) criterion between the vectorial inputs [14]. Due to the robust performance under a wide range of circumstances, it is common to use the aggregated statistics (Fig. 4) defined as follows:

$$d_{(i,j)} = \sum_{(g,h)\in\varsigma} f_{DS}\left(\mathbf{x}_{(i,j)}, \mathbf{x}_{(g,h)}\right) \qquad (3)$$

where $f_{DS}(.,.)$ measures the distance or similarity between the vectorial inputs $\mathbf{x}_{(i,j)}$ and $\mathbf{x}_{(g,h)}$.

It has been shown in [15] that the distance/similarity aggregation concept defined over the input vectors is suitable for many color image processing applications, such as noise filtering, edge detection, microarray image processing, spectral interpolation, spatial interpolation, television and digitized artwork image enhancement, etc. Note that the vectorial nature of the color inputs provides a high degree of freedom in designing the distance/similarity measures suitable for the aforementioned tasks. Numerous designs including Euclidean distance, Minkowski metric, and sophisticated similarity measures are discussed in detail in [2],[14],[15].

It is not difficult to see that the application of data-adaptive filtering concept to the scalar signals such as the gray-scale images reduces (3) to the aggregated absolute differences between the scalar inputs. Assuming for the simplicity $f(.)$ equivalent to the identity function keeping the input unchanged, the direct implication of the scalar data-adaptive filtering concept to color images results in the so-called component-wise processing [16]:

$$x_{(r,s)k} = \sum_{(i,j)\in\varsigma} w'_{(i,j)} x_{(i,j)k} \qquad (4)$$

where the k-th component $x_{(r,s)k}$ of the color vector $\mathbf{x}_{(r,s)}$ is estimated using the k-th components $x_{(i,j)k}$ of the spatially neighbored color vectors $\mathbf{x}_{(i,j)}$, for $(i,j) \in \varsigma$ and $k = 1, 2, 3$. Analogously, the weighting coefficients $w_{(i,j)}$ are determined using the function of the aggregated absolute differences $d_{(i,j)}$ between the color components (Fig. 4):
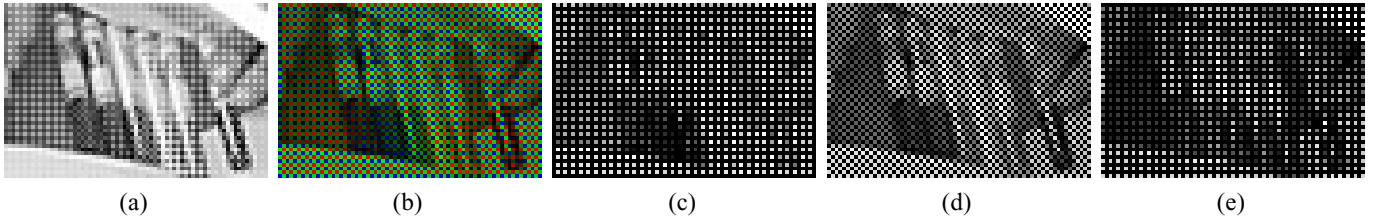
(a)   (b)   (c)   (d)   (e)

Fig. 5. Demosaicking input: (a) raw sensor data, (b) CFA data arranged as a color image, (c-e) R,G,B channels separated from the image shown in (b).

$$d_{(i,j)} = \sum_{(g,h)\in\varsigma} \left| x_{(i,j)k} - x_{(g,h)k} \right| \qquad (5)$$

However, it is widely observed that the application of scalar filters, such as the one defined in (4), on each color channel separately disrupts the correlation that exists between the color components of natural images represented in a correlated color space. Since each processing step is usually accompanied by a certain inaccuracy, the formation of the output color vector from the separately processed color components usually produces color artifacts [15]. Therefore, vector filtering techniques that treat the color image as a vector field are more appropriate for most, if not all, color image processing tasks.

Unfortunately, the fact that the CFA image contains only a single measurement in each spatial location limits the applicability of the vectorial processing. To overcome this limitation, the proposed demosaicking framework, described in the sequence, employs any color model of [17],[18],[19], [20] to utilize the spectral characteristics of the CFA inputs and decrease the level of artifacts present in the demosaicked image [8],[21],[22]. To restore the camera output in a sharp manner, the framework utilizes the aggregated statistics of (5) as the base for the edge-sensing mechanism. Since these statistics can be mapped into the weighting coefficients in many ways and similarly the design characteristics of the solution are further directed by the use of any color model, numerous demosaicking solutions can be designed within the proposed framework. Thus, the framework has a great potential to fully satisfy various needs of the end-user.

III. PROPOSED DEMOSAICKING FRAMEWORK

The CCD/CMOS sensor is a monochromatic device and thus, the raw CFA sensor values should be arranged as a $K_1 \times K_2$ gray-scale image $z : Z^2 \to Z$ representing a two-dimensional matrix of integer samples $z_{(r,s)}$ with $r = 1, 2, ..., K_1$ denoting the image row and $s = 1, 2, ..., K_2$ indicating the image column. Since the arrangement of the color filters in the CFA is known from the camera manufacturers, these raw sensor data can be re-arranged to a $K_1 \times K_2$ color (RGB) image $\mathbf{x} : Z^2 \to Z^3$ representing a two-dimensional matrix of three-component samples [19]. Assuming the commonly accepted Bayer pattern with a GRGR phase in the first row (Fig. 3), the gray-scale pixels $z_{(r,s)}$ (Fig. 5a) can be transformed into the RGB vectors $\mathbf{x}_{(r,s)} = [x_{(r,s)1}, x_{(r,s)2}, x_{(r,s)3}]$ with $x_{(r,s)k}$ indicating the R ($k = 1$), G ($k = 2$) and B ($k = 3$) component, as follows [19],[20]:

$$\mathbf{x}_{(r,s)} = \begin{cases} [z_{(r,s)}, 0, 0] & \text{for } r \text{ odd and } s \text{ even,} \\ [0, 0, z_{(r,s)}] & \text{for } r \text{ even and } s \text{ odd,} \\ [0, z_{(r,s)}, 0] & \text{otherwise.} \end{cases} \qquad (6)$$

Since only a single measurement $x_{(r,s)k}$ varying in k is available in each spatial location (r,s), color vectors $\mathbf{x}_{(r,s)}$ of the image $\mathbf{x}$ (Fig. 5b) are completed using two zero components (Fig. 5c-e). The missing components are interpolated taking advantages of an edge-sensing mechanism and a spectral model employed in the demosaicking procedure. These key elements used to construct the proposed framework are described in the sequence.

A. Edge-sensing mechanism

Through the use of the weighting coefficients the edge-sensing mechanism is employed to direct the interpolation process along the edges [11],[21],[23]. It is common to track the changes in the structural content of the image using some form of inverse gradients [8],[24]. The rationale behind this choice is rather simple: the small gradients reflect the inputs localized along the edge, whereas the large gradients correspond to the directions defined across the edge. Assuming the weighted averaging operator, such as the one used in (1) or (4), the weighting coefficients equivalent to inversely proportional values of the gradients are thus used to proportionally represent the contribution of each input. The inputs associated with the large gradients are appropriately penalized through the corresponding small weights and the inputs corresponding to the small gradients (large weights) contribute mostly to the output.

Employing the aggregated statistics $d_{(i,j)}$ obtained in (5), the weighting coefficients can be expressed as follows [14]:

$$w_{(i,j)} = f(d_{(i,j)}) \qquad (7)$$

where $f(.)$ is the function mapping the aggregated absolute differences between the inputs to the weighting coefficients. The actual shape of the function $f(.)$ determines the properties of the weights, [14].

Taking advantages of inversely proportional gradients, probably the simplest form of $f(.)$ is defined as follows [14]:

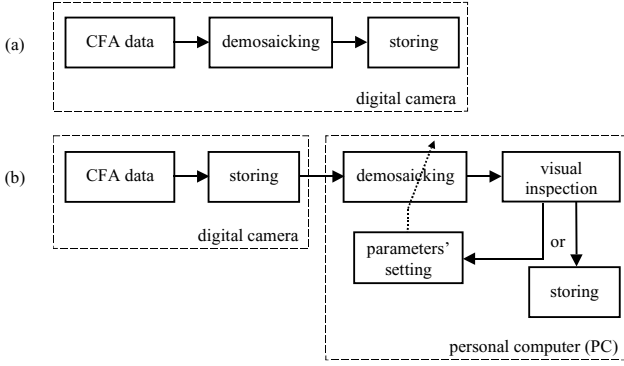$$w_{(i,j)} = \frac{1}{1 + d_{(i,j)}} \qquad (8)$$

Fig. 6. Demosaicking performed: (a) in the fully automated manner required in most digital cameras, (b) using the PC-based interface. In the application scenario the end-user controls: (a) storing/deleting of the restored images, (b) storing/deleting of both raw CFA images and the images restored using the PC, and also controls the interpolation process by setting the parameters of the demosaicking algorithm.

where the constant is added in the denominator to avoid the singularities in calculations. Due to the simplicity of (8), this implementation is suitable to be embedded in the conventional digital cameras (Fig. 6a) requiring: i) the real-time performance and high processing rate, and ii) no user control of the interpolation process.

However, many digital cameras store the captured images in the raw (CFA) format. In this case it is expected that: i) the demosaicking is guided using the personal computer, ii) the end-user has the possibility to control the interpolation process by setting the parameters, and iii) based on the quality of the restored output the end-user has an option to eventually re-run the demosaicking with other setting of the parameters. Following this application scenario (Fig. 6b), more sophisticated forms of $f(.)$ are necessary.

It is known [14] that the relationship between distances measured in physical units and perception is generally exponential. Therefore, high visual quality of the reconstructed images can be obtained using the generalized form of a sigmoidal function [14]:

$$w_{(i,j)} = \frac{\lambda}{\left(1 + \exp(d_{(i,j)})\right)^{\tau}} \qquad (9)$$

where τ is a parameter adjusting the weighting effect of the membership function and λ is a normalizing constant.

Both (8) and (9) are used here to demonstrate the potential of the proposed framework, which can be designed to perform fully automated demosaicking as well as the interactive processing of the raw sensor data. It is not difficult to see that various forms of $f(.)$ can be used instead (8) and (9) which makes the proposed framework very flexible.

B. Spectral modeling based interpolator

It is well-known that natural RGB images exhibit a strong spectral correlation between the color channels [1],[2]. This is the reason that many powerful demosaicking solutions

developed in the past incorporate the spectral image characteristics to the interpolation process and thus reduce the amount of the spectral problems (color shifts, artifacts, moire, etc.) present in the demosaicked output [7],[8],[11],[21],[23]. To the date, four spectral models suitable for this task have been introduced. They include the color-ratio model (CRM) [17], the color-difference model (CDM) [18], the normalized CRM utilizing the linear shifting operations (NRSM) [19],[20] and the normalized CRM utilizing both linear scaling and shifting operations (NRSSM) [20]. The common base for these designs consists in the use of the RG or BG spectral characteristics. Since natural color images do not have large areas with uniform spectral characteristics, the smoothing operations are performed over the spectral inputs available in the spatial neighborhood ς.

Let $\oplus$ be a relation between the two inputs A and B, and $\overline{\oplus}$ be an inverse operator to $\oplus$. Assuming that $\oplus$ and $\overline{\oplus}$ denote the following operations:

$$A \oplus B = A/B \quad \text{and} \quad A \overline{\oplus} B = AB \qquad (10)$$
$$A \oplus B = A - B \quad \text{and} \quad A \overline{\oplus} B = A + B \qquad (11)$$
$$A \oplus B = (A+\beta)/(B+\beta) \quad \text{and} \quad A \overline{\oplus} B = (A+\beta)B - \beta \qquad (12)$$
$$A \oplus B = (\alpha A+\beta)/(\alpha B+\beta) \quad \text{and} \quad A \overline{\oplus} B = [(\alpha A+\beta)B - \beta]/\alpha \qquad (13)$$

the operators $\oplus$ and $\overline{\oplus}$ defined in (10), (11), (12) and (13) generalize the operations defined behind the CRM, CDM, NRSM and NRSSM, respectively. The parameters β and α denote the linear shifting and scaling operations and allow for the end-user to obtain the additional control of the interpolation process.

Combining the generalized spectral model and the data-adaptive filtering concept of (4), the following two forms of the CFA interpolator are obtained:

$$x_{(r,s)k} = x_{(r,s)2} \overline{\oplus} \sum_{(i,j)\in\varsigma} \left(w'_{(i,j)}(x_{(i,j)k} \oplus x_{(i,j)2}) \right) \qquad (14)$$

$$x_{(r,s)2} = x_{(r,s)k} \overline{\oplus} \sum_{(i,j)\in\varsigma} \left(w'_{(i,j)}(x_{(i,j)2} \oplus x_{(i,j)k}) \right) \qquad (15)$$

The use of $x_{(r,s)2}$ in (14) denotes the available G component placed at an interpolated location (r,s). The neighboring locations $(i,j)\in\varsigma$ correspond to the available G components $x_{(i,j)2}$ and R (for $k=1$) or B (for $k=3$) components $x_{(i,j)k}$ used to interpolate the R or B component $x_{(r,s)k}$. Analogously, the G component $x_{(r,s)2}$ of (15) is obtained using the R ($k=1$) or B ($k=3$) component $x_{(r,s)k}$ located at the interpolation location (r,s) and the surrounding G components $x_{(i,j)2}$ and R (or B) components $x_{(i,j)k}$ available in spatially neighboring positions $(i,j)\in\varsigma$.

Since the data-adaptive filtering concept is defined here over the spectral quantities $x_{(i,j)k} \oplus x_{(i,j)2}$ in (14) or $x_{(i,j)2} \oplus x_{(i,j)k}$ in (15), both $x_{(r,s)2}$ in (14) and $x_{(r,s)k}$ in (15) are used to normalize the operand from the modeling to the pixel domain.
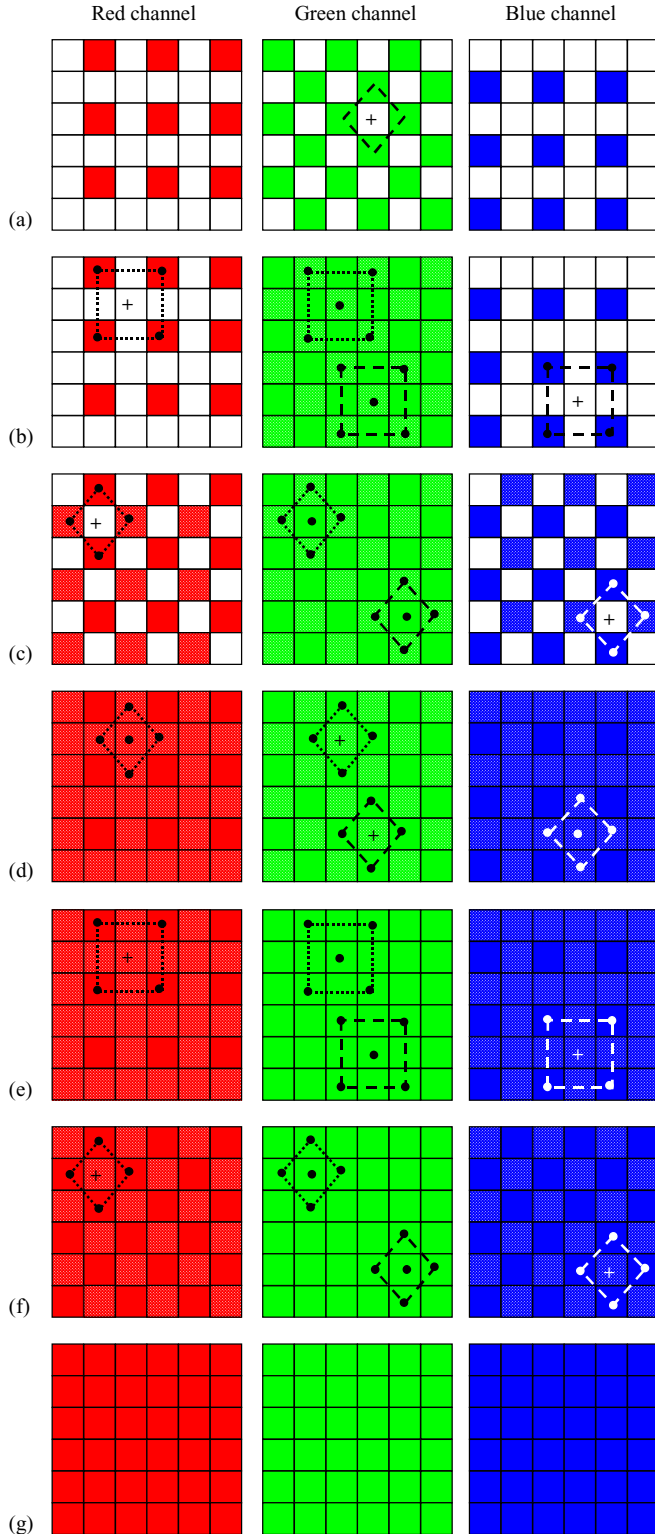
Red channel   Green channel   Blue channel

(a)
(b)
(c)
(d)
(e)
(f)
(g)

Fig. 7. The proposed demosaicking framework is designed to operate on four-neighbor spatial arrangements: (a,c,d,f) a diamond-shape mask $\varsigma=\{(r-1,s),(r,s-1),(r,s+1),(r+1,s)\}$, (b,e) a square-shape mask $\varsigma=\{(r-1,s-1),(r-1,s+1),(r+1,s-1),(r+1,s)\}$. The individual processing steps correspond to: (a) component-wise G channel interpolation, (b,c) spectral model based interpolation, (d-f) spectral model based correction of previously interpolated components, (g) the restored image. Symbols ● and + denote the components used in the interpolator's input and the components being interpolated, respectively.

Input:

> $K_1 \times K_2$ Bayer CFA image z equivalent to raw sensor (grayscale) data.
> Specify the spectral model used in the algorithmic steps (14) and (15).
> Specify the function $f(.)$ in (7) used to map the aggregated statistics (5) into the edge-sensing weights used to control the procedure.
> Set the parameters (if any) used to control the edge-sensing mechanism and/or the spectral model.

Demosaicking procedure:

1. Re-arrange $K_1 \times K_2$ image z to $K_1 \times K_2$ color image $\mathbf{x}$ using (6).
2. Let $\varsigma = \{(r-1,s),(r,s-1),(r,s+1),(r+1,s)\}$, Fig. 7a. Calculate the aggregated statistics of (5) with $k=2$. Interpolate the missing G component using (4) with $k=2$.
2. Let $\varsigma = \{(r-1,s-1),(r-1,s+1),(r+1,s-1),(r+1,s+1)\}$, Fig. 7b. Calculate the aggregated statistics of (5) with $k=1$ for R components. Interpolate the missing R components using (14) and $k=1$. Use $k=3$ in (5) and (14) to obtain the missing B components.
3. Let $\varsigma = \{(r-1,s),(r,s-1),(r,s+1),(r+1,s)\}$, Fig. 7c. Calculate the aggregated statistics of (5) with $k=1$ for R components. Interpolate the missing R components using (14) and $k=1$. Use $k=3$ in (5) and (14) to obtain the missing B components.
4. Let $\varsigma = \{(r-1,s),(r,s-1),(r,s+1),(r+1,s)\}$, Fig. 7d. Calculate the aggregated statistics of (5) using $k=2$. Correct the interpolated G components using (15) with $k=1$ for R CFA locations and $k=3$ for B CFA locations.
5. Let $\varsigma = \{(r-1,s-1),(r-1,s+1),(r+1,s-1),(r+1,s+1)\}$, Fig. 7e. Calculate the aggregated statistics of (5) with $k=1$ for R components. Correct the interpolated R components using (14) and $k=1$. Use $k=3$ in (5) and (14) to correct the interpolated B components.
6. Let $\varsigma = \{(r-1,s),(r,s-1),(r,s+1),(r+1,s)\}$, Fig. 7f. Calculate the aggregated statistics of (5) with $k=1$ for R components. Correct the interpolated R components using (14) and $k=1$. Use $k=3$ in (5) and (14) to correct the interpolated B components.

Output:

> $K_1 \times K_2$ restored, full color output image.

Fig. 8. Algorithm of the proposed demosaicking framework.

C. Demosaicking procedure

Since the G color plane of the color CFA image $\mathbf{x}$ obtained using (6) contains double number of the components than R or B color planes (Fig. 7a) and thus also contains a major part of spatial information of the image being reconstructed, most, if not all, demosaicking algorithms start the demosaicking procedure with interpolating the missing G components $x_{(r,s)2}$. These are located at the center (r,s) of four spatially surrounding G components $x_{(i,j)2}$, for $(i,j) \in \varsigma$, forming a diamond shape $\varsigma = \{(r-1,s),(r,s-1),(r,s+1),(r+1,s)\}$ on the image lattice (Fig. 7a). The application of the data-adaptive concept here is straightforward: the components $x_{(r,s)2}$ are obtained using (4) with $k=2$ and the weights are defined through the aggregated statistics of (5) with $k=2$.

The availability of the fully populated G color plane allows to utilize more information in interpolating the missing R and B components. This is realized using the spectral model based data-adaptive filtering concept of (14). As shown in Fig. 7b, the available R (for $k=1$) or B (for $k=3$) components $x_{(i,j)k}$
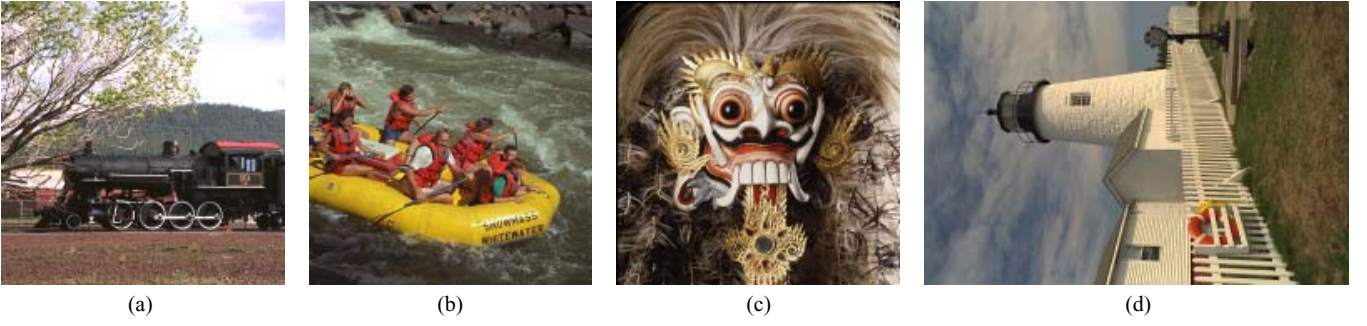
Fig. 9. Test color images: (a) Train, (b) Rafting, (c) Mask, (d) Lighthouse.

form a square- shape mask $\varsigma = \{(r-1,s-1),(r-1,s+1),(r+1,s-1),(r+1,s+1)\}$. The missing R (or B) component $x_{(r,s)k}$ located at the center of ς is obtained using these available R (or B) components $x_{(i,j)k}$ and G components $x_{(.,.)2}$ located at the neighboring positions $(i,j) \in \varsigma$ and the interpolated location (r,s). It is not difficult to see that this interpolation step produces the patterns (Fig. 7c) with the missing R (or B) components located at the center of the diamond-shape mask. Therefore, (14) with $\varsigma = \{(r-1,s),(r,s-1),(r,s+1),(r+1,s)\}$ is repeated to produce fully populated R and B color planes.

Although this step results in the full color output, the use of correction [11],[23], or postprocessing steps [19],[24],[25], improves the image sharpness and further reduces the amount of color artifacts present in the demosaicked output. Since G components were previously interpolated without any spectral information using (4), the utilization of more information coming from both G and R (or G and B) color planes in the spectral model-based variant (15) of the data-adaptive filter greatly boosts the performance of the proposed framework. If the G component $x_{(r,s)2}$ being corrected is located at the original R CFA locations (r,s), then (15) with $k=1$ is used. Otherwise, (15) with $k=3$ is used to correct the G components located at the original B CFA location. In both cases a diamond shape mask $\varsigma = \{(r-1,s),(r,s-1),(r,s+1),(r+1,s)\}$ defines the location of the original G components (Fig. 7d).

The proposed demosaicking procedure completes with the correction of previously interpolated R and B components. At this processing stage the corrected G color plane and the original R (or B) CFA components described by $\varsigma = \{(r-1,s-1),(r-1,s+1),(r+1,s-1),(r+1,s+1)\}$ (Fig. 7e) are used to improve the contrast in the R (or B) channel. Similarly to the interpolation phase, the repetition of (14) with $\varsigma = \{(r-1,s),(r,s-1),(r,s+1),(r+1,s)\}$ is necessary to complete the correction tasks (Fig. 7f).

The algorithmic steps performed by the proposed CFA demosaicking framework are summarized, in pseudo-code format, in Fig. 8. As it can be seen from this overview, apart from the selection of the spectral model and the edge-sensing mapping, the end-user can have an additional option to control the demosaicking procedure by setting the parameters of these two construction elements. Depending on the actual application, the proposed framework can be used to perform either an automatic demosaicking [26] or the required visual quality is obtained by tuning the parameters [27].
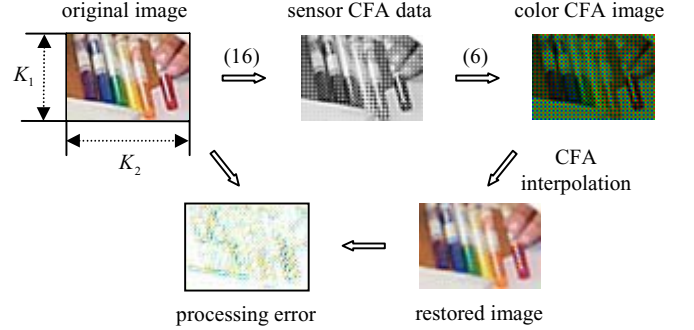

Fig. 10. Evaluation procedure.

IV. EXPERIMENTAL RESULTS

To examine performance of the proposed framework, a number of color images have been used. The examples such as the 512×512 images Train, Rafting, and Mask, and the 768×512 image Lighthouse are depicted in Fig. 9. These test images have been captured using three-sensor devices and normalized to 8-bit per channel RGB representation.

Fig. 10 shows the evaluation procedure commonly used in the research community [9],[11],[20],[21],[28],[29]. The set of the test images, such as those shown in Fig. 9, is constituted. Mosaic versions of the $K_1 \times K_2$ original images **o** are created by discarding color information in a GRGR phased Bayer CFA filter (Fig. 3) resulting in the raw, sensor (grayscale) CFA image z with the pixels $z_{(r,s)}$ defined as follows [11]:

$$z_{(r,s)} = \begin{cases} o_{(r,s)1} & \text{for } r \text{ odd and } s \text{ even,} \\ o_{(r,s)3} & \text{for } r \text{ even and } s \text{ odd,} \\ o_{(r,s)2} & \text{otherwise.} \end{cases} \quad (16)$$

where $o_{(r,s)k}$ denote the R ($k=1$), G ($k=2$) and B ($k=3$) component of the original color vector $\mathbf{o}_{(r,s)} = [o_{(r,s)1}, o_{(r,s)2}, o_{(r,s)3}]$ with $r = 1,2,...,K_1$ and $s = 1,2,...,K_2$. The color version $\mathbf{x}$ of the CFA image z is generated through (6) and the restored (demosaicked, output) images are obtained from $\mathbf{x}$ applying any CFA interpolation method such as the numerous solutions designed within the proposed framework. The efficiency of the interpolation methods is measured, in terms of both objective and subjective criteria, by comparing the quality of the output, restored image against the original image.

TABLE I
COMPARISON OF THE METHODS USING THE TRAIN IMAGE

Method	MAE	MSE	NCD
AP	3.499	51.6	0.0560
API	5.037	117.7	0.0776
BI	10.569	466.4	0.1450
BD	4.622	85.8	0.0744
C2D2	4.884	113.0	0.0702
EMI	5.795	170.6	0.0845
KA	4.562	91.8	0.0710
MFI	4.873	109.9	0.0798
PVM	6.780	214.3	0.0958
SAI	5.985	166.9	0.0866
SHT	7.759	257.7	0.1085
Proposed framework:			
(8) + CRM	4.437	79.7	0.0726
(8) + CDM	3.392	49.5	0.0536
(8) + NRSM	3.448	51.5	0.0544
(8) + NRSSM	3.399	49.6	0.0538
(9) + CRM	4.448	79.6	0.0728
(9) + CDM	3.423	49.8	0.0542
(9) + NRSM	3.477	51.9	0.0550
(9) + NRSSM	3.429	49.4	0.0544

TABLE II
COMPARISON OF THE METHODS USING THE RAFTING IMAGE

Method	MAE	MSE	NCD
AP	2.014	20.6	0.0358
API	2.273	21.5	0.0399
BI	4.796	92.2	0.0753
BD	2.288	21.0	0.0396
C2D2	2.303	22.5	0.0375
EMI	3.002	45.3	0.0492
KA	3.066	55.7	0.0504
MFI	2.410	25.0	0.0434
PVM	3.103	40.7	0.0504
SAI	3.030	41.5	0.0507
SHT	3.875	61.1	0.0626
Proposed framework:			
(8) + CRM	3.193	57.4	0.0547
(8) + CDM	2.007	19.1	0.0351
(8) + NRSM	1.993	18.4	0.0344
(8) + NRSSM	2.003	19.0	0.0350
(9) + CRM	3.225	57.5	0.0552
(9) + CDM	2.074	20.1	0.0361
(9) + NRSM	2.048	19.3	0.0353
(9) + NRSSM	2.068	20.0	0.0360

In this paper, the mean absolute error (MAE), the mean square error (MSE) and the normalized color difference (NCD) criterion are used to objectively measure the performance of the demosaicking algorithms. Assuming that $\mathbf{o}_{(r,s)} = [o_{(r,s)1}, o_{(r,s)2}, o_{(r,s)3}]$ denotes the pixel in a $K_1 \times K_2$ original RGB image $\mathbf{o}$ and $\mathbf{x}_{(r,s)} = [x_{(r,s)1}, x_{(r,s)2}, x_{(r,s)3}]$ is the pixel in a $K_1 \times K_2$ restored output $\mathbf{x}$, the MAE and MSE are defined as follows:

$$\text{MAE} = \frac{1}{3K_1K_2} \sum_{k=1}^{3} \sum_{r=1}^{K_1} \sum_{s=1}^{K_2} \left| o_{(r,s)k} - x_{(r,s)k} \right| \tag{17}$$

$$\text{MSE} = \frac{1}{3K_1K_2} \sum_{k=1}^{3} \sum_{r=1}^{K_1} \sum_{s=1}^{K_2} \left(o_{(r,s)k} - x_{(r,s)k} \right)^2 \tag{18}$$

Since the MAE and MSE criteria do not measure the difference between the two images in the units corresponding to human perception, some other measures, such as the NCD criterion, defined in perceptually uniform color spaces, are necessary. The NCD values are obtained as follows [14]:

$$\text{NCD} = \frac{\sum_{r=1}^{K_1} \sum_{s=1}^{K_2} \sqrt{\sum_{k=1}^{3} \left(o'_{(r,s)k} - x'_{(r,s)k} \right)^2}}{\sum_{r=1}^{K_1} \sum_{s=1}^{K_2} \sqrt{\sum_{k=1}^{3} \left(o'_{(r,s)k} \right)^2}} \tag{19}$$

where $\mathbf{o}'_{(r,s)} = [o'_{(r,s)1}, o'_{(r,s)2}, o'_{(r,s)3}]$ and $\mathbf{x}'_{(r,s)} = [x'_{(r,s)1}, x'_{(r,s)2}, x'_{(r,s)3}]$ are the vectors representing the RGB vectors $\mathbf{o}_{(r,s)}$ and $\mathbf{x}_{(r,s)}$, respectively, in the CIE LUV color space [3] with the white point D65. To obtain the LUV representation of $\mathbf{o}_{(r,s)}$ and $\mathbf{x}_{(r,s)}$, the procedure requires to transform the RGB values to their XYZ equivalents, which are further transformed to the LUV values. After these operations, the color difference can be quantified using (19).

The solutions designed within the proposed framework are compared, in terms of performance, against the well-known Bayer pattern based demosaicking schemes. The prior-art methods used here for the comparisons are the alternative projection (AP) scheme [9], the adaptive color plane interpolation (API) scheme [30], the bilinear interpolation scheme [28], the bilinear difference (BD) interpolation scheme [29], the color correlation-directional derivatives (C2D2) scheme [21], the edge map interpolation (EMI) scheme [31], the Kimmel's algorithm (KA) [23], the median filter interpolation (MFI) scheme [32], the principle vector method (PVM) [33], the saturation based adaptive interpolation (SAI) scheme [34] and the smooth hue transition (SHT) scheme [17].

To boost efficiency of the proposed framework, $\beta = 256$ and $\alpha = 0.05$ for NRSSM, and $\beta = 256$ for NRSM, are used as the default setting of the normalized spectral models [19],[20]. Following the findings listed in [27], $\lambda = 1$ and $\tau = 0.005$ are used as the default setting of (9). Note that using the generalized form of the spectral interpolator (14) and (15), the framework unifies our previous developments based on (8) and (11) [26], and (9) and (11) [27].

Tables I-IV summarize the objective results obtained using the test images shown in Fig. 9. As it can be observed the conventional BI scheme introduces significant inaccuracy into the restoration process. Other techniques, such as API, BD, EMI, MFI, SAI and SHT, which utilize more advanced interpolators compared to the BI scheme, provide better results in terms of all objective criteria. Simple inspection of the values listed in Tables I-IV suggests that the solutions designed within the proposed framework outperform other demosaicking schemes including sophisticated C2D2, KA, and PVM schemes. The proposed framework and the AP approach achieve the best performance in terms of all image quality measures. However, if the image contains a number of
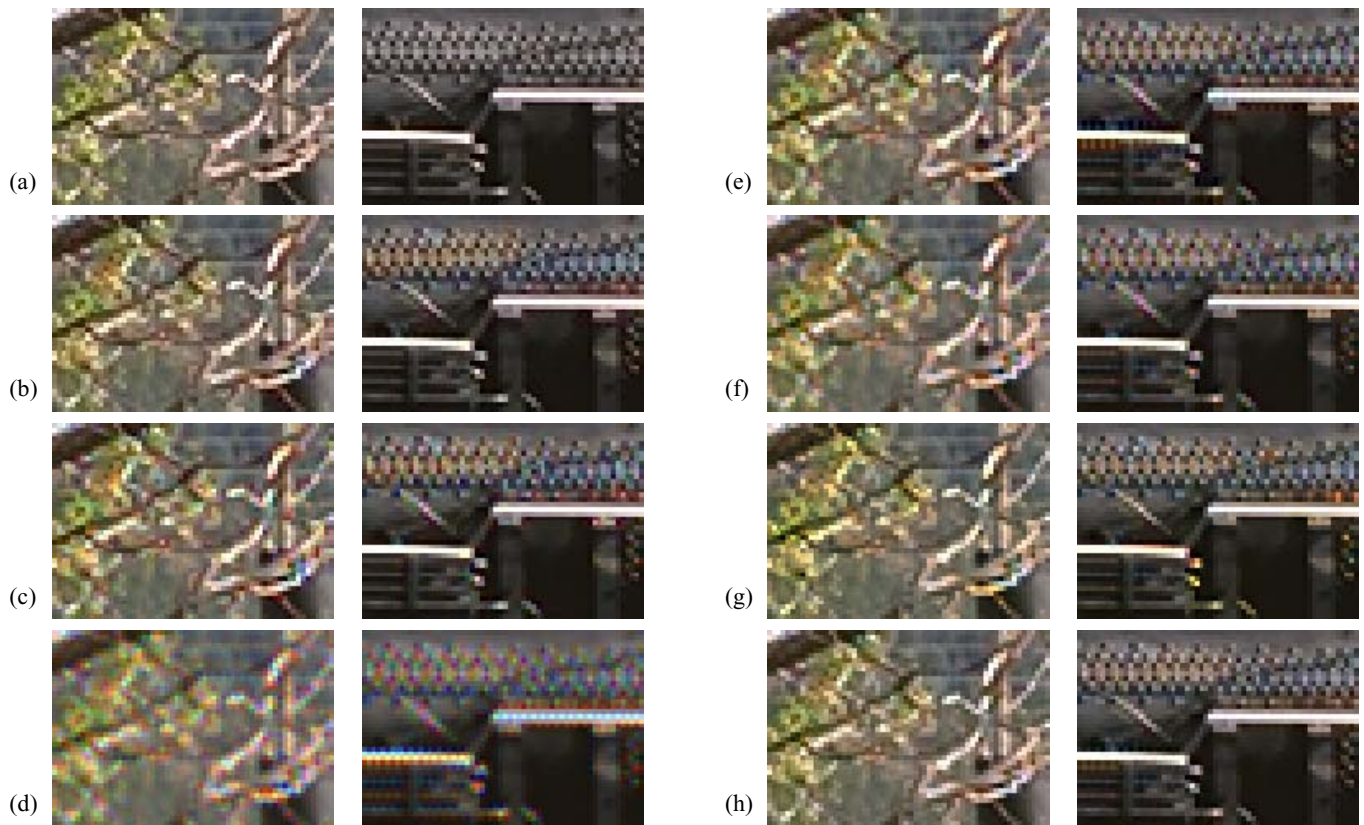
Fig. 11. Enlarged parts of the original Train image (a) and the images restored using: (b) AP, (c) API, (d) BI, (e) BD, (f) C2D2, (g) KA, (h) the proposed framework.
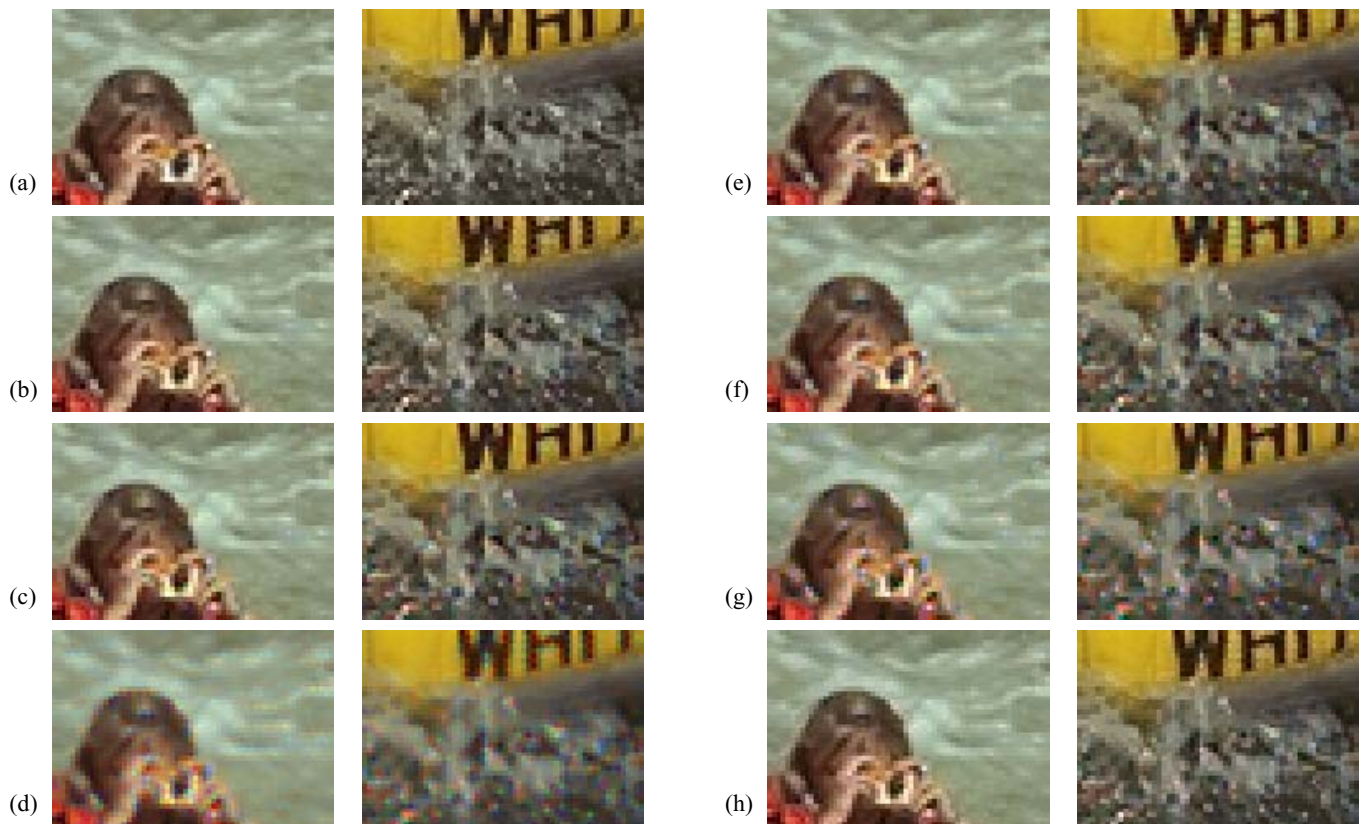

Fig. 12. Enlarged parts of the original Rafting image (a) and the images restored using: (b) AP, (c) API, (d) BI, (e) BD, (f) MFI, (g) PVM, (h) the proposed framework.
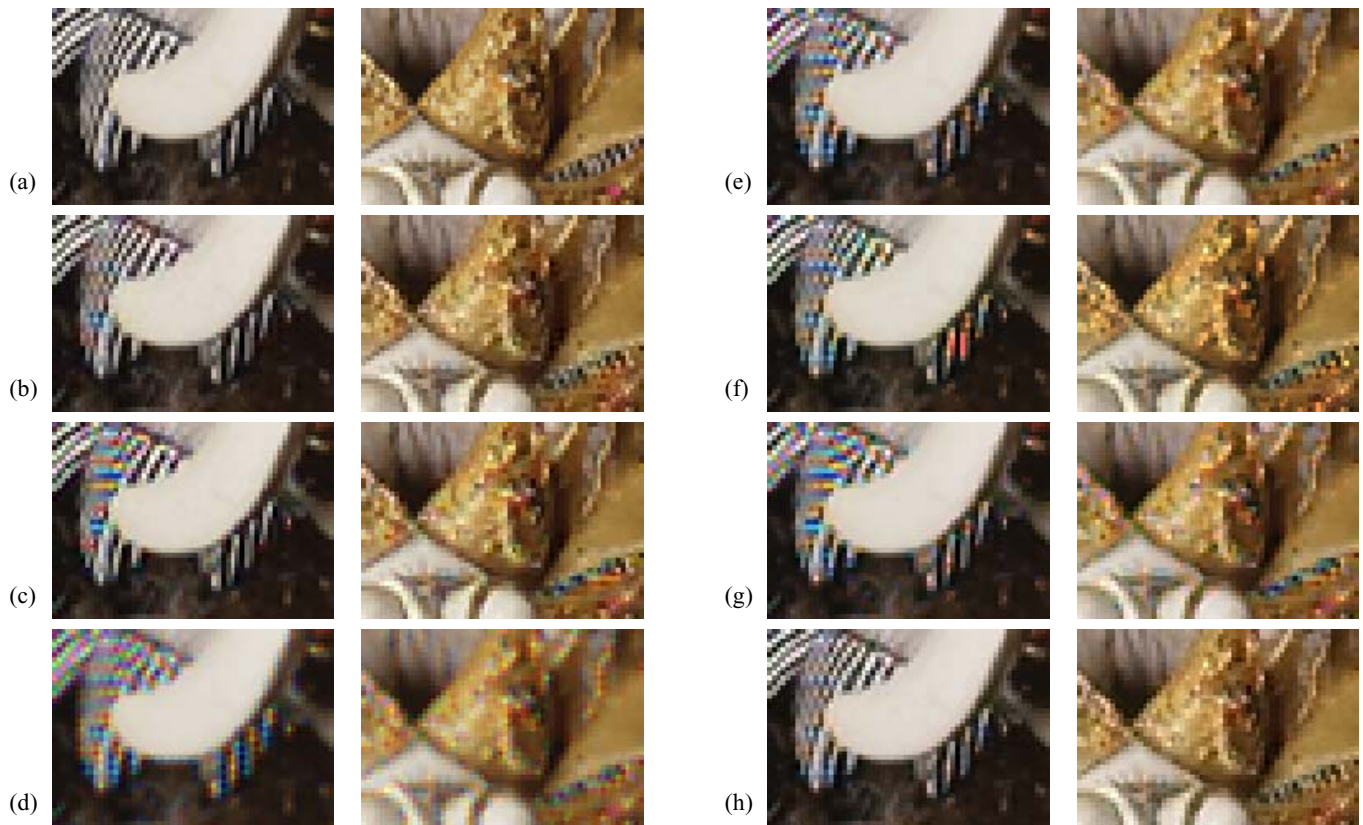
Fig. 13. Enlarged parts of the original Mask image (a) and the images restored using: (b) AP, (c) API, (d) BI, (e) C2D2, (f) KA, (g) PVM, (h) the proposed framework.
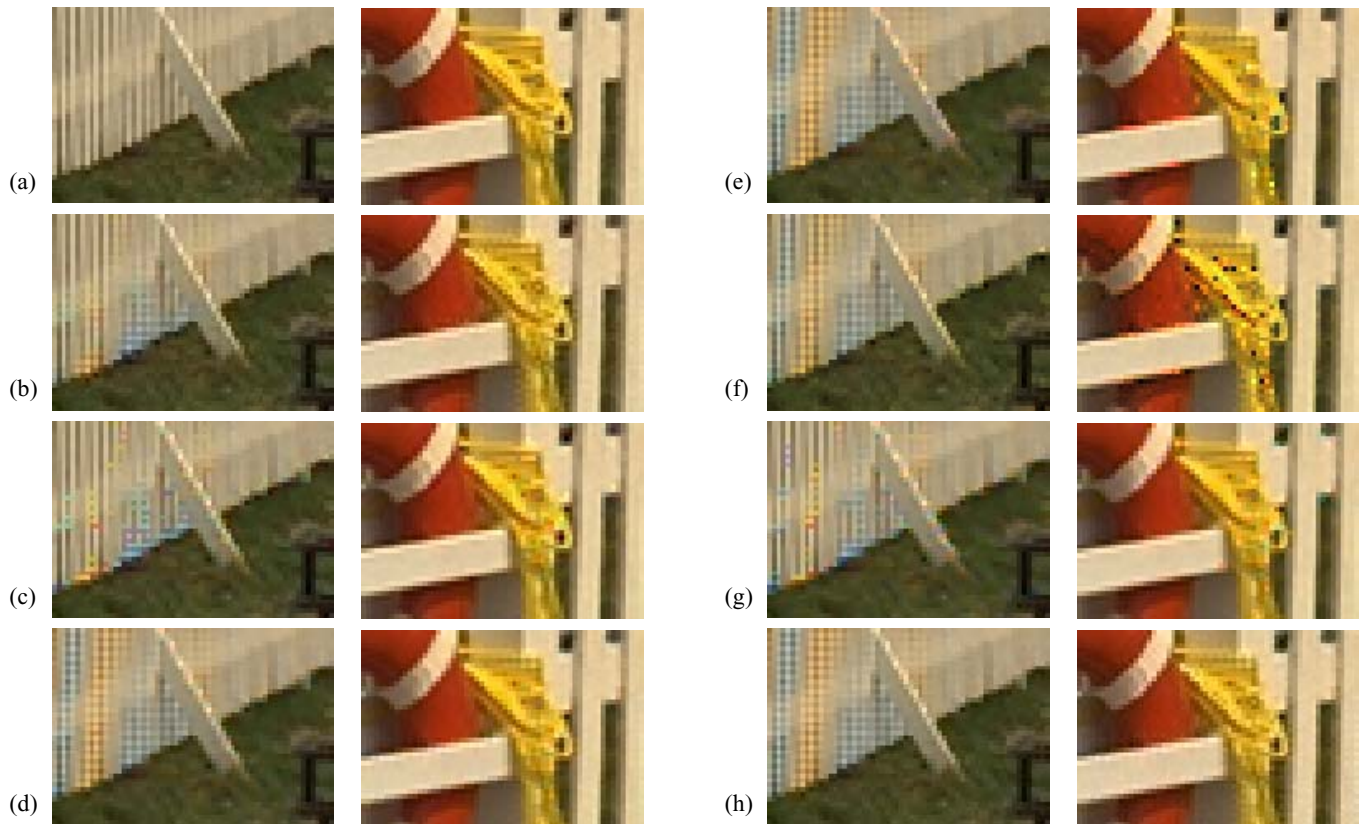

Fig. 14. Enlarged parts of the original Lighthouse image (a) and the images restored using: (b) AP, (c) API, (d) C2D2, (e) EMI, (f) KA, (g) PVM, (h) the proposed framework.

TABLE III
COMPARISON OF THE METHODS USING THE MASK IMAGE

Method	MAE	MSE	NCD
AP	3.434	42.3	0.0754
API	3.852	57.6	0.0830
BI	7.376	212.9	0.1328
BD	3.817	49.2	0.0819
C2D2	3.753	56.3	0.0750
EMI	4.424	94.8	0.0825
KA	3.836	60.2	0.0736
MFI	3.829	57.3	0.0822
PVM	5.012	101.1	0.0984
SAI	4.482	87.4	0.0854
SHT	5.867	131.7	0.1070
Proposed framework:			
(8) + CRM	4.194	65.6	0.0815
(8) + CDM	3.334	40.0	0.0729
(8) + NRSM	3.372	41.1	0.0711
(8) + NRSSM	3.336	40.0	0.0728
(9) + CRM	4.235	66.4	0.0822
(9) + CDM	3.387	40.9	0.0744
(9) + NRSM	3.427	42.1	0.0724
(9) + NRSSM	3.388	40.8	0.0742

TABLE IV
COMPARISON OF THE METHODS USING THE LIGHTHOUSE IMAGE

Method	MAE	MSE	NCD
AP	1.467	7.3	0.0237
API	1.817	12.7	0.0298
BI	4.468	108.3	0.0653
BD	2.149	19.7	0.0321
C2D2	2.005	16.2	0.0301
EMI	2.322	23.1	0.0370
KA	2.241	19.4	0.0365
MFI	2.341	31.8	0.0380
PVM	2.364	23.5	0.0366
SAI	2.570	24.3	0.0403
SHT	3.619	58.5	0.0522
Proposed framework:			
(8) + CRM	2.589	25.0	0.0414
(8) + CDM	1.712	12.0	0.0264
(8) + NRSM	1.716	12.0	0.0264
(8) + NRSSM	1.717	12.0	0.0265
(9) + CRM	2.603	25.3	0.0415
(9) + CDM	1.741	12.1	0.0281
(9) + NRSM	1.741	12.5	0.0267
(9) + NRSSM	1.723	12.4	0.0267

TABLE V
COMPUTATIONAL COMPLEXITY OF THE PROPOSED FRAMEWORK
EXPRESSED THROUGH THE NUMBER OF NORMALIZED OPERATIONS

Interpolator / Criterion	ADs	SBs	MUs	DIs	AVs
(4) + (8)	18	6	4	8	6
(14 or 15) + (8) + CRM	18	6	5	12	6
(14 or 15) + (8) + CDM	19	10	4	8	6
(14 or 15) + (8) + NRSM	27	7	5	12	6
(14 or 15) + (8) + NRSSM	27	7	14	13	6

fine details varying in color, the use of the proposed framework can take advantage over the AP scheme and decrease the amount of the spatial and spectral artifacts to a level corresponding to the lowest error values (Tables I-III).

Figs. 11-14 facilitate the visual comparison of enlarged parts of both the original images and the restored outputs obtained using the methods under consideration. The results indicate that the BI, EMI and SHT schemes blur image edges and delete both structural content and fine details. The API, BD, C2D2, MFI, KA and PVM approaches exhibit improved detail-preserving characteristics. However, the corresponding outputs contain a number of color artifacts as it can be seen by simply contrasting these results to the original images. On the other hand, the proposed method and AP approach preserve the image details and reduce the amount of color artifacts. The employed edge-sensing mechanism in conjunction with the correction steps and spectral modeling help the proposed algorithm to accurately restore the image and to preserve the original structural content in most cases. Fig. 14 shows that the proposed framework is not capable to eliminate aliasing errors in the picket fence area of the image Lighthouse to a level obtained using the AP, API, or PVM schemes. However, other examples depicted in Figs. 11-14 clearly demonstrate that in all other image regions, with various color-appearance and complexity of the structural content, the proposed framework outperforms other solutions and clearly produces visually pleasing, naturally colored images.

Apart from the numerical behavior (actual performance) of any algorithm, its computational complexity is a realistic measure of its practicality and usefulness. Therefore, the proposed demosaicking framework is analyzed here in terms of normalized operations, such as additions (ADs), subtractions (SUs), multiplications (MUs), divisions (DIs), absolute values (AVs) and exponents (EXs). Table V summarizes the total number of operations per interpolated location denoted for the individual proposed interpolators. Note that the use of (9) instead (8) requires 4 EXs in addition to the amounts listed in Table V. The analysis suggests that the proposed framework represents an efficient and cost-effective demosaicking solution. Moreover, since a number of operations can be implemented in parallel and shared forms, additional computational saving may occur.

The execution of the developed demosaicking tool on a PC with an Intel Pentium IV 2.40 GHz CPU, 512 MB RAM, Windows XP operating system and MS Visual C++ 5.0 programming environment, took (on average) 0.87 sec per a 512 × 512 input image processed by the fully automated solutions, e.g. those based on (8) and some spectral model.

Summarizing the results presented above, the following conclusions can be drawn: i) the use of a data-filtering concept based generalized edge-adaptive spectral interpolator and postprocessor constitutes a new, powerful demosaicking framework for Bayer CFA based imaging devices, ii) both fully automated and user-controlled solutions designed within the proposed framework produce visually pleasing restored images, iii) the proposed demosaicking framework is robust and achieves significant improvement in terms of the image quality evaluation compared to the state-of-the-art, and iv) the proposed framework is computationally efficient and it can be relatively easily implemented either in software or hardware.

V. CONCLUSION

A new demosaicking framework operating on Bayer CFA data was introduced. The framework utilizes an efficient data-adaptive filtering concept, a generalized spectral model, and a correction step to produce visually pleasing restored color images. At the same time it yields excellent results in terms of commonly used objective image quality criteria.

REFERENCES

[1] G. Wyszecki, and W. S. Stiles, *Color Science, Concepts and Methods, Quantitative Data and Formulas*. John Wiley, N.Y., 2nd Edition, 1982.

[2] K. N. Plataniotis and A. N. Venetsanopoulos, *Color Image Processing and Applications*. Springer Verlag, Berlin, 2000.

[3] G. Sharma and H. J. Trussell, "Digital color imaging," *IEEE Transactions on Image Processing*, vol. 6, no. 7, pp. 901-932, July 1997.

[4] J. Adams, K. Parulski, and K. Spaulding, "Color processing in digital cameras," *IEEE Micro*, vol. 18, no. 6, pp. 20-30, Nov./Dec. 1998.

[5] S. Yamanaka, "Solid state camera," *U.S. Patent 4 054 906*, Nov. 1977.

[6] B. E. Bayer, "Color imaging array," *U.S. Patent 3 971 065*, July 1976.

[7] R. Ramanath, W. E. Snyder, G. L. Bilbro, and W. A. Sander III, "Demosaicking methods for Bayer color arrays," *Journal of Electronic Imaging*, vol. 11, no. 3, pp. 306-315, July 2002.

[8] L. Chang and Y. P. Tang, "Effective use of spatial and spectral correlations for color filter array demosaicking," *IEEE Transactions on Consumer Electronics*, vol. 50, no. 1, pp. 355-365, May 2004.

[9] B. Gunturk, Y. Altunbasak, and R. Mersereau, "Color plane interpolation using alternating projections," *IEEE Transactions on Image Processing*, vol. 11, no.9, pp. 997-1013, September 2002.

[10] T. Sakamoto, C. Nakanishi, and T. Hase, "Software pixel interpolation for digital still cameras suitable for a 32-bit MCU," *IEEE Transactions on Consumer Electronics*, vol. 44, no. 4, pp. 1342-1352, Nov. 1998.

[11] R. Lukac, K. N. Plataniotis, D. Hatzinakos, and M. Aleksic, "A novel cost effective demosaicing approach," *IEEE Transactions on Consumer Electronics*, vol. 50, no. 1, pp. 256-261, February 2004.

[12] H. J. Trussell and R. E. Hartwig, "Mathematics for demosaicking," *IEEE Trans. Image Processing*, vol. 11, no. 4, pp. 485-492, April 2002.

[13] R. Lukac, K. Martin, and K. N. Plataniotis, "Digital camera zooming based on unified CFA image processing steps," *IEEE Transactions on Consumer Electronics*, vol. 50, no. 1, February 2004.

[14] K. N. Plataniotis, D. Androutsos, and A. N. Venetsanopoulos, "Adaptive fuzzy systems for multichannel signal processing," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1601-1622, September 1999.

[15] R. Lukac, B. Smolka, K. Martin, K. N. Plataniotis, and A. N. Venetsanopoulos, "Vector filtering for color imaging," *IEEE Signal Processing Magazine, Special Issue on Color Image Processing*, vol. 22, no. 1, January 2005.

[16] J. Zheng, K. P. Valavanis, and J. M. Gauch, "Noise removal from color images," *J. Intelligent and Robotic Systems*, vol. 7, pp. 257-285, 1993.

[17] D. R. Cok, "Signal processing method and apparatus for producing interpolated chrominance values in a sampled color image signal," *US Patent 4 642 678*, February 1987.

[18] J. Adams, "Design of practical color filter array interpolation algorithms for digital cameras," *Proc. SPIE*, vol. 3028, pp. 117-125, Feb. 1997.

[19] R. Lukac, K. Martin, and K. N. Plataniotis, "Demosaicked image post-processing using local color ratios," *IEEE Transactions on Circuit and Systems for Video Technology*, vol. 14, no. 6, pp. 914-920, June 2004.

[20] R. Lukac, and K. N. Plataniotis, "Normalized color-ratio modelling for CFA interpolation," *IEEE Transactions on Consumer Electronics*, vol. 50, no. 2, pp. 737-745, May 2004.

[21] N. Kehtarnavaz, H. J Oh, and Y. Yoo, "Color filter array interpolation using color correlations and directional derivatives," *Journal of Electronic Imaging*, vol. 12, no. 4, pp. 621-632, October 2003.

[22] R. Lukac, K. Martin, and K. N. Plataniotis, "Colour-difference based demosaicked image postprocessing," *IEE Electronics Letters*, vol. 39, no. 25, pp. 1805-1806, December 2003.

[23] R. Kimmel, "Demosaicing: image reconstruction from color CCD samples," *IEEE Transactions on Image Processing*, vol. 8, no. 9, pp. 1221-1228, September 1999.

[24] W. Lu and Y. P. Tang, "Color filter array demosaicking: new method and performance measures," *IEEE Transactions on Image Processing*, vol. 12, no. 10, pp. 1194-1210, October 2003.

[25] C. C. Koh, J. Mukherjee, and S. K. Mitra, "New efficient methods of image compression in digital cameras with color filter array," *IEEE Trans. Consumer Electronics*, vol. 49, no. 4, pp. 1448-1456, Nov. 2003.

[26] R. Lukac and K. N. Plataniotis, "An efficient CFA interpolation solution," in *Proc. 46th International Symposium focused on Navigation, Multimedia and Marine in Zadar, Croatia*, pp. 543-548, June 2004.

[27] R. Lukac, K. Martin, K. N. Plataniotis, and B. Smolka, "Bayer pattern demosaicking using data-dependent adaptive filters," in *Proc. 22nd Biennial Symposium on Communications in Kingston, Canada*, pp. 207-209, June 2004.

[28] P. Longere, X. Zhang, P. B. Delahunt, and D. H. Brainard, "Perceptual assessment of demosaicing algorithm performance," *Proceedings of the IEEE*, vol. 90, no. 1, pp. 123-132, January 2002.

[29] S. C. Pei and I.K. Tam, "Effective color interpolation in CCD color filter arrays using signal correlation," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 13, no. 6, pp. 503-513, June 2003.

[30] J. F. Hamilton and J. E. Adams, "Adaptive color plane interpolation in single sensor color electronic camera," *U.S. Patent 5 629 734*, 1997.

[31] B. S. Hur and M. G. Kang, "High definition color interpolation scheme for progressive scan CCD image sensor," *IEEE Trans. Consumer Electronics*, vol. 47, no. 2, pp. 179-186, February 2001.

[32] W. T. Freeman, "Median filter for reconstructing missing color samples," *U.S. Patent 4 724 395*, February 1988.

[33] R. Kakarala and Z. Baharav, "Adaptive demosaicing with the principal vector method," *IEEE Transactions on Consumer Electronics*, vol. 48, pp. 932-937, November 2002.

[34] C. Cai, T. H. Yu, and S. K. Mitra, "Saturation-based adaptive inverse gradient interpolation for Bayer pattern images," *IEE Proc. - Vision, Image, Signal Processing*, vol. 148, no. 3, pp. 202-208, June 2001.

Rastislav Lukac received the M.S. (Ing.) and Ph.D. degrees in Telecommunications from the Technical University of Kosice, Slovak Republic in 1998 and 2001, respectively. From February 2001 to August 2002 he was an Assistant Professor at the Department of Electronics and Multimedia Communications at the Technical University of Kosice. Since August 2002 he is a Researcher in Slovak Image Processing Center in Dobsina, Slovak Republic. From January 2003 to March 2003 he was a Postdoctoral Fellow at the Artificial Intelligence & Information Analysis Lab at the Aristotle University of Thessaloniki, Greece. Since May 2003 he has been a Post-doctoral Fellow with the Edward S. Rogers Sr. Department of Electrical and Computer Engineering at the University of Toronto in Toronto, Canada.

His research interests include digital camera image processing, microarray image processing, multimedia security, and nonlinear filtering and analysis techniques for color image & video processing. Dr. Lukac is a Member of the IEEE Signal Processing Society. In 2003 he was awarded the NATO/NSERC Science Award.

Konstantinos N. Plataniotis received the B. Engineering degree in Computer Engineering from the Department of Computer Engineering and Informatics, University of Patras, Patras, Greece in 1988 and the M.S and Ph.D degrees in Electrical Engineering from the Florida Institute of Technology (Florida Tech), Melbourne, Florida in 1992 and 1994 respectively. He was affiliated with the Computer Technology Institute (C.T.I), Patras, Greece from 1989 to 1991. From August 1997 to June 1999 he was an Assistant Professor with the School of Computer Science at Ryerson University. He is currently an Assistant Professor at the Edward S. Rogers Sr. Department of Electrical & Computer Engineering where he researches and teaches adaptive systems and multimedia signal processing.

Dr. Plataniotis is a Senior Member of IEEE, a past member of the IEEE Technical Committee on Neural Networks for Signal Processing, and the Technical Co-Chair of the Canadian Conference on Electrical and Computer Engineering (CCECE) 2001, and CCECE 2004.