UNIVERSITÀ DI PADOVA  FACOLTÀ DI INGEGNERIA

Color Image Reconstruction for Digital Cameras

Ph.D. THESIS

Author: Daniele Menon

Director: Ch.mo Prof. Matteo Bertocco
Supervisor: Ch.mo Prof. Giancarlo Calvagno

2008

DEPARTMENT OF INFORMATION ENGINEERING
UNIVERSITY OF PADOVA

PH. D. SCHOOL ON INFORMATION ENGINEERING
COURSE ON ELECTRONIC AND TELECOMMUNICATION ENGINEERING

UNIVERSITÀ DI PADOVA  FACOLTÀ DI INGEGNERIA

Dipartimento di Ingegneria dell'Informazione

Scuola di Dottorato di Ricerca in Ingegneria dell'Informazione
Indirizzo: Ingegneria Elettronica e delle Telecomunicazioni

CICLO XXI

COLOR IMAGE RECONSTRUCTION FOR DIGITAL CAMERAS

Direttore della Scuola: Ch.mo Prof. Matteo Bertocco

Supervisore: Ch.mo Prof. Giancarlo Calvagno

Dottorando: Daniele Menon

Abstract

Recently we observed a fast diffusion of digital cameras that are able to acquire images and videos directly in the digital format. This new acquisition technique allowed to explore new strategies to process, save and display images and videos.

Digital cameras require many operations to process the data acquired by the sensor. In this thesis I present an overview of the techniques used in practical realizations and proposed in the literature. Particular attention is paid to the algorithms that are more connected to the image processing area.

Among them, the process that is the most important for the quality of the resulting images and the most computational demanding is *demosaicking*. This consists in the reconstruction of the full color representation of an image from the data acquired by a sensor provided with a *color filter array* that in each pixel acquires a color component only instead of the three values that are necessary to represent a color image. The most common color filter array is called Bayer pattern, from the name of his inventor Bryce Bayer. In this thesis an overview of the demosaicking techniques presented in the literature is given and three new methods that allow to obtain good performances with a reduced computational cost are proposed. The first two are based on directional interpolations and are made adaptive to the image behavior through analysis of the edges and wavelet transformations. The last proposed technique, instead, is based on regularization methods, an useful tool to find a solution for an ill-conditioned inverse problem.

Since the sensor introduces a noisy component in the acquired data, an algorithm to perform demosaicking and denoising jointly is also analyzed. It exploits wavelet transformations.

Finally, a method to adaptively interpolate the image is presented, in order to increase the resolution and improve the visual quality of the details in the image. This technique is based on an analysis of the statistical local behavior of the image.

Sommario

Negli ultimi anni abbiamo assistito ad una rapida diffusione di fotocamere e telecamere in grado di acquisire immagini e video direttamente in formato digitale. Questo nuovo tipo di acquisizione ha dato la possibilità di esplorare nuove strategie per l'elaborazione, l'archiviazione e la visualizzazione delle immagini e dei video.

Le fotocamere (e telecamere) digitali richiedono un elevato numero di operazioni per elaborare i dati acquisiti dal sensore. In questa tesi viene offerta una panoramica delle tecniche utilizzate nella pratica e di quelle presentate nella letteratura scientifica. In particolare, si è data particolare attenzione agli algoritmi maggiormente legati all'area dell'elaborazione dei segnali.

Tra questi, il procedimento più importante per la qualità finale dell'immagine e più impegnativo per le risorse computazionali delle fotocamere è senza dubbio il *demosaicking* (traducibile con *demosaicizzazione*). Esso consiste nella ricostruzione della rappresentazione a colori dell'immagine a partire dai dati acquisiti con un *color filter array* che, in ogni pixel, preleva una sola componente di colore invece delle tre necessarie per la visualizzazione di un'immagine a colori. Un *color filter array* molto diffuso è il Bayer pattern, così chiamato dal nome del suo inventore Bryce Bayer. All'interno della tesi viene fornita un'ampia panoramica delle strategie proposte per il demosaicking nella letteratura scientifica e, inoltre, vengono presentati tre nuovi approcci in grado di fornire ottime prestazioni nonostante il loro ridotto costo computazionale. I primi due sono basati su interpolazioni direzionali e resi *adattivi* al contenuto dell'immagine attraverso opportune analisi del comportamento dei bordi e delle trasformazioni *wavelet*. Un'altra tecnica proposta, invece, è basata sui metodi di regolarizzazione, utili per trovare soluzioni in problemi matematicamente non invertibili.

Dal momento che i sensori aggiungono una componente rumorosa ai dati acquisiti, viene analizzato anche un algoritmo che permette di effettuare congiuntamente demosaicking e denoising utilizzando delle trasformazioni *wavelet*.

In conclusione, viene presentato un metodo per effettuare un'interpolazione adattiva dei dati in modo da aumentare la risoluzione e migliorare la visione dei dettagli. Tale tecnica si basa su un'analisi del comportamento statistico locale dell'immagine.

To the masters Gian Antonio Mian and Luca Manai,
with never-ending gratefulness.

Acknowledgments

During my Ph.D. course, I received help and support from many people. I would like to thank all of them and I hope to not forget anybody.

First of all, I would like to thank my supervisor, Giancarlo Calvagno. He introduced me to signal and image processing and gave me a continuous support, especially during troubles. I am very grateful also to Luciano Sbaiz for giving me the opportunity to spend a period of research at the École Politechnique Fédérale de Lausanne. Moreover, I shared with him a lot of inspiring discussions that are very useful to my research. Thanks also to prof. Martin Vetterli and prof. Sabine Süsstrunk that allowed me to carry on my research in their laboratory in Lausanne.

Special thanks must be made to my colleagues of the Digital Image and Signal Laboratory of the University of Padova, Lorenzo, Stefano, Simone, Degiu and Ottavio. They made my work environment peculiar and stimulating and tolerated my frequent questions. Amongst them, I would thank Stefano (my first "coadvisor") for the snowballs in Munich and my flatmate Lorenzo for avoiding to wake me up with cold water. I include also the other Ph.D. students of the department, that shared with me these three years. Moreover, I cannot forget the support received during my stay in Lausanne from Fabrizio and from my colleagues of the Audiovisual Communication Laboratory.

I would like to thank also prof. Sebastiano Battiato for reading and reviewing my thesis and all the reaserchers that provided me with their software. Amongst then I am grateful to Lei Zhang for sharing professional knowledge.

I must remember all the friends in Bassano and Padova, for whose comments made in my master thesis are still effective and sound.

I thank also my parents, my brother Carlo and my sister Elena, with whom I spend my week-ends in Bassano. Last but not least, a lot of thanks to Giulia for her support and patience.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

In the last decade, we observed a considerable diffusion of digital image usage in the everyday life, and consumer electronic devices able to capture digital images, such as digital cameras, PDA, cell phones, etc., have become very popular. Moreover, streaming video websites such as YouTube have simplified the process of sharing images and videos. The diffusion of digital images has also a strong social impact, since new consumer products have become available to meet new social needs. For instance, videosurveillance permits to keep big areas under control, improving public safety. Many medical applications based on image processing increase the capability of medicine for diagnostics and various other purposes.

This large diffusion has become possible thanks to the intense development of color image acquisition technology. In fact, several electronic devices capable to capture digital images (and video-sequences) are now available to many applications. Their reduced cost allow them to become popular not only among photography fans but also among average users.

Despite narrowing profit margins and increasing competition, the growing ubiquity of digital multimedia drives the demands for improved image quality in color imaging devices, altough often improvements are expected not to be followed by a price rise.

Therefore, industry keeps on investing in development of better image acquisition strategies. The research carried out in this field involves signal processing solutions. In fact, several signal processing procedures have proved to be able to offer attractive benefits and cost savings that cannot be ignored.

From an academic viewpoint, it is interesting to analyze these emerging problems, providing new efficient solutions, and propose strategies which can be able to address new perspectives for the industrial world.

In this thesis I analyze some open problems in the image acquisition chain of a digital camera (or video-camera). Different solutions are proposed. Many of them can be efficiently applied in current digital cameras to provide high-quality and cost-effective performances. Other parts of this thesis intend to exploit some signal processing approaches to design novel solutions capable to improve the present state of the art.

The next section briefly describes the image acquisition scheme of a typical digital camera,

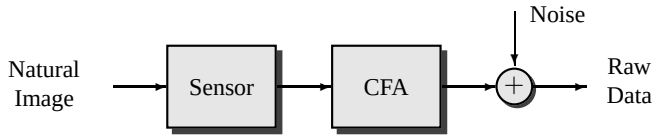


Figure 1.1: *Image acquisition in a digital camera.*

reviewing a number of steps or modules that comprise a camera pipeline. Section 1.3 explains which parts of this pipeline will be object of discussion in the thesis and which solutions will be proposed.

1.2 Color digital cameras

Fig. 1.1 shows the acquisition process of a natural image from a digital camera. Natural environments emit or reflect photons that are captured by the photosensitive elements of the camera. The photon signal is a continuous function of space and wavelengths, but the sensors of the camera acquires a discrete function both in the space (along a grid of pixels) and in wavelengths, according to the spectral sensitivity of the sensors [7]. Therefore, aliasing occurs when the spatial frequencies of the captured scene are finer than what the spacing between pixels in the sensor can support. Without preventive actions, low-frequency *moiré* pattern will show up in the high-frequency regions. As a mean to reduce this phenomenon, many cameras use an antialiasing filter, applying one of the two approaches: using a birefringent material that exploits polarization to spread a beam of light over multiple pixels, or using phase delay techniques that shift neighboring rays out of phase, consequently attenuating high-frequency content. Control mechanisms interact with the sensors to determine the exposure (aperture size, shutter speed, and automatic gain control) and the focal position of the lens. Moreover, it is conventional to include an infrared blocking filter, called a *hot mirror*, to reflect infrared energy, because the silicon substrate used in the sensor is sensitive to the near-infrared part of the spectrum [120]. An alternative to the hot mirror is to use a filter made of an optical material absorbing infrared radiation. However, if these filters are exposed to too much infrared energy without proper cooling, they can crack or shatter. Therefore many manufacturers prefer a hot mirror solution [4].

The key-element of the digital camera is the sensor array that converts light into electrical signal. Many different image sensor architectures are available, and the most common solutions employ *Charged-Coupled Devices* (CCD) or *Complementary Metal Oxide Silicon*) CMOS sensors.

Because there is the need to measure three color bands at each grid locations, this requires the use of more than one sensor in each pixel. A more robust and cheaper solution is to place a color filter array (CFA) on top of the sensor element. Of the many CFA patterns available, the most popular is the Bayer pattern (see Chapter 2).

Moreover, image acquisition introduces an additive noise term. The noise can be assumed uncorrelated with the image, but this assumption can result inadequate for many devices because the noise is almost always dependent on the acquired signal [45, 61]. In particular, a good

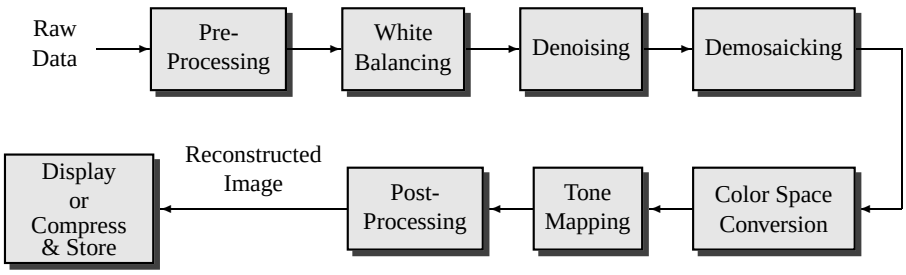


Figure 1.2: *Pipeline for the image reconstruction in a typical digital camera.*

noise model is given by a signal-dependent Poissonian part, modeling the photon sensing, and a Gaussian part, for the remaining stationary disturbances in the output data.

The resulting data obtained from the sensor, usually called *raw data*, need to be processed before producing a pleasing representation of the captured scene. The signal flowchart shown in Fig. 1.2 briefly summarizes the necessary processing pipeline. It should be noted that the sequence of operation differs from manufacturer to manufacturer [67, 92, 120].

Now, we describe each block in Fig. 1.2:

- **Pre-processing.** The acquired data present some drawbacks that, if not corrected, can affect the resulting image after the reconstruction. A first problem is that, even with the lens cap on, a dark current signal is recorded, which is due to thermally generated electrons in the sensor substrate. Usually this effect is compensated after an estimation of the intensity due to dark current. Another necessary operation is the *flare compensation* because light entering the optics of the camera gets scattered and reflected, causing a nonuniform shift in the measured energy, in particular in images where a bright source of light is in the field of view [120].

- **White Balancing.** The human visual system has the ability to map "white" colors to the sensations of white, even though an object has different radiance when it is illuminated with different light sources. Also a digital camera needs to match the acquired image with the colors perceived by humans, using an estimation of the scene illuminant. Often no knowledge about the illuminant is available and it has to be estimated from the raw data. For instance, a naive white balancing approach assumes that a white patch must induce maximal camera responses in the three channels, then the illuminant is estimated by the maximum values $R_{\max}$, $G_{\max}$, and $B_{\max}$ of the three color channels R, G, and B, respectively. The white-balanced image has components given by $R/R_{\max}$, $G/G_{\max}$, $B/B_{\max}$. However often this estimation is not accurate enough. An overview of more efficient techniques is presented in [10]. White balancing can take place also after demosaicking or concurrently with the color space conversion [67].

- **Denoising.** Many procedures to remove the noise introduced during the acquisition has been proposed [17, 45, 73]. If denoising is performed before demosaicking (as in Fig. 1.2), it has to be applied on each acquired color component separately, otherwise it can be carried out on the full color image after demosaicking. However, it should be

considered that demosaicking can modify the characteristics of the noise, making the denoising procedure more complex. Moreover, demosaicking a noisy image is more difficult than demosaicking a denoised one.

- **Demosaicking.** Demosaicking is the process of reconstructing the full color image from the data acquired with the CFA. Its implementation depends on the CFA pattern used by the camera. It is, by far, the most computationally intensive step in the processing pipeline [120]. All demosaicking techniques make use of pixel neighborhood to estimate the values of the pixels colors that were not measured. Moreover, many approaches apply adaptive methods to improve the quality of edges and detail in the reconstructed image. The solutions proposed for demosaicking will be described in the Chapter 2 of this thesis.

- **Color Space Conversion.** Because the color coordinates defined by the spectral sensitivity functions of the sensors do not correspond exactly to the standardized color space (such as the sRGB space), a color space conversion has to be applied. It consists in a change of basis, with a pixel-wise multiplication. Additional color space conversions may be required for image compression, which usually operate in an opponent color space (such as the YUV space) [92].

- **Tone Mapping.** It is the operation that matches scene to display luminances, in order to reproduce visually pleasing images that correspond to the expectation of the observer. Usually it involves a nonlinear function. The most famous tone mapping is the so-called *gamma correction*, consisting in a power law, but more sophisticated tone mapping methods exist. The function parameters can depend on global characteristics of the image [122], or alternatively the output pixel values can be obtained on the basis of the pixel surround in order to change local features, such as increasing the local contrast to improve detail visibility [123]. Tone mapping can be performed also before demosaicking [100].

- **Post-processing.** Many camera manufacturers apply different approaches to improve the quality of the resulting image. For instance, a deblurring technique can be employed to reduce the blurring effect given by the sensors. Moreover, since the human visual system is particularly sensitive to the reconstruction of the edges, many digital camera apply edge enhancement methods such as unsharp masking. Sometimes, a *coring* technique is used to remove detail information that has no significant contribution to image detail and behaves much like noise [120]. Some digital cameras with a low sensor resolution employ a digital zoom to interpolate the image and improve the vision.

- **Display or Compress & Store.** Finally, the image can be displayed. Otherwise it can be compressed for storage purposes. Consumer digital cameras tend to use the EXIF format or the JPEG compression. On the other hand, professional cameras, which have the freedom of much larger sensors and storage space, prefer to give also the possibility to store the image in an uncompressed format or exploiting a lossless compression. With these high-quality cameras it is also possible to store directly the raw data. In this way, the several operations we described in this section are carried out off-line, exploiting

many computational resources, in order to obtain a more pleasing reconstruction of the images.

1.3 Thesis overview

As described in the previous section, one of the most important steps during the acquisition of digital images is demosaicking. Differently from other tasks of Fig. 1.2, it involves both spatial and color analysis. Usually it is the most computational demanding procedure of the digital camera pipeline, resulting the bottleneck of the whole framework. It is also the part where signal processing can give the most significant contribution. Therefore, Chapters 2-5 are devoted to this issue. In particular, Chapter 2 describes the arrangements introduced for the CFA and presents an overview of the existing demosaicking approaches.

Chapter 3 analyzes a demosaicking method based on directional filtering that we proposed in [p4]. This approach achieves high-quality images with a reduced computational cost, candidating itself for implementation in simple low-cost cameras or in video capture devices with high values of resolution and frame-rate. The same chapter describes also a post-processing algorithm to improve the quality of the demosaicked images, presented in [p2].

Chapter 4 introduces a wavelet-based demosaicking technique, where the properties of the wavelet transform are exploited to obtain a reliable reconstruction also of the most detailed regions of the image. A wavelet analysis drives an adaptive interpolation of the available data. Part of this chapter has been presented in [p5].

Instead, Chapter 5, gives an overview of the regularization theory, then exploits this framework to design a reconstruction approach where demosacking is considered as an inverse problem. It takes into account also the blurring effect and the noise introduced by the sensors. A preliminary version of this chapter has been presented in [p6] and analyzed in [p8].

As described in the previous section, the effects of demosaicking and denoising are strictly related. Denoising directly the CFA image in order to apply demosaicking on a denoised image is not simple and also to remove the noise after demosaicking has relevant drawbacks. Therefore, Chapter 6 proposes a new technique where the two procedures are considered jointly.

In many digital camera often an increase of the resolution of the acquired image is desired. Chapter 7 describes an approach for image magnification, exploiting some local statistical properties of the image. In this way, also small details have a better representation. This technique is analyzed also in [p7].

Chapter 2

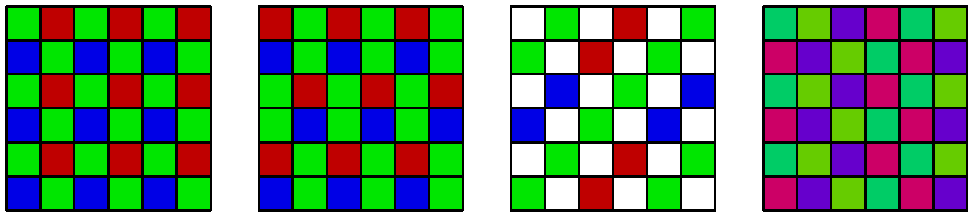
Overview on demosaicking

This chapter presents a survey of the demosaicking techniques proposed in the literature. First, the need of a color filter array in a digital camera is discussed, then several demosaicking approaches are presented and analyzed in detail.

2.1 Introduction

In a typical digital camera each pixel is equipped with a sensor and a microlens. The photons collected by the microlens reach the photosensitive element of the sensor, usually a CMOS (*Complementary Metal Oxide Silicon*) or a CCD (*Charge-Coupled Device*). The CMOS photo diode active pixel sensors measure the intensity of the light using a photo diode and three transistors, while the CCD sensors rely on the electron-hole pair that is generated when a photon strikes silicon [67].

In a color digital camera the acquisition may require three sensors for pixel, each of them sensible to a particular wavelength, however the positioning of the sensors is not straightforward. A first solution is to project the light entering the camera onto each spectral sensor, but this solution is expensive and leads to some phase delay between the components. Another approach is to stack the color sensors on top of one another, as done in the Foveon cameras, but the exposure times are high because the light has to penetrate three levels of silicon. Therefore, most of the digital cameras use a single-sensor technique, with a grid of different color sensors, called *Color Filter Array* (CFA). The most common pattern for the CFA was introduced by Bayer [12], and it samples the green band using a quincunx grid, while red and blue are obtained by a rectangular grid, as shown in Fig. 2.1(a). In this way, the density of the green samples is twice than that of the red and blue channels. The Bayer pattern can be applied also using the subtractive primaries cyan, magenta, and yellow (CMY), that have the advantage of being more sensitive to light, as in the Kodak DCS620x digital camera. Another CFA configuration has been used by Sony in the DSC F828 digital camera, exploiting also the Emerald filter, while many companies use a cyan, magenta, yellow, and green array (CMYG), to provide a compromise between maximum light sensitivity and high color quality (such as in the Nikon Coolpix 990). Moreover, an array consisting in yellow, cyan, green, and panchromatic sensors originated at Hitachi is used by JVC in some of its video cameras. Several other arrangements



(a) (b) (c) (d)

Figure 2.1: *Four arrangements for the Color Filter Arrays: (a) the Bayer pattern [12]; (b) the pattern proposed in [93] by Lukac and Plataniotis; (c) the pattern proposed by Kodak [56]; (d) one of the patterns proposed in [69] by Hirakawa and Wolfe.*

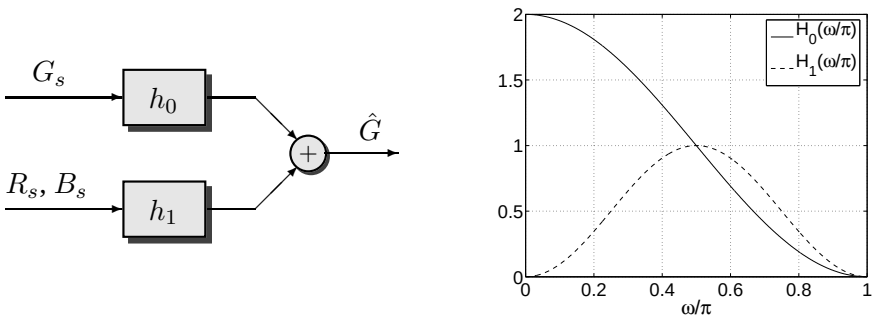
have been proposed in the literature, such as the pattern shown in Figure 2.1(b), presented in [93] or the new scheme introduced by Kodak [56], involving panchromatic pixels as in Figure 2.1(c). Recently Hirakawa and Wolfe proposed new configurations where the sensors in each pixel capture a linear combination of the traditional red, green and blue components, in order to reduce the aliasing given by the sampling of the image [69]. One of these new patterns is shown in Figure 2.1(d).

Due to the subsampling of the color components, an interpolation step is required in order to estimate the missing values and reconstruct a full color representation of the image. This process is called *demosaicking*, and has a large influence on the final image quality. Avoiding the introduction of visible artifacts such as *moiré* and *zipper* effects is desirable.

In literature, many demosaicking algorithms have been presented. As a macter of fact, the classical image interpolation methods, such as nearest-neighbor replication, bilinear interpolation and cubic spline interpolation, are not able to exploit the information given by the three color components jointly and do not provide good performances. Therefore better results are achieved by reconstruction approaches specifically designed for the reconstruction of images sampled with a CFA.

In this chapter we review most of the techniques considered for demosaicking. Since a huge number of approaches has been proposed, we subdivide them in five groups, namely heuristic methods, algorithms based on directional interpolations, frequency-domain approaches, wavelet-based methods, and reconstruction techniques. Moreover, we describe also some approaches proposed to perform demosaicking jointly with super-resolution, and algorithms for joint demosaicking and denoising. Although a separation into these categories could seem subjective, we think that it will be useful to read and follow this chapter, without being thrown into confusion by the high number of the proposed algorithms. Other overviews about demosaicking complementary to this chapter are given in [11, 53, 84].

In the following we concentrate our discussion on the reconstruction from data acquired with the Bayer patter, that is the most used CFA arrangement. However, in general many of the demosaicking algorithms developed for the Bayer pattern can be extended to other CFA configurations.



(a) (b)

Figure 2.2: *(a) Reconstruction of the green signal; (b) Frequency response of the two filters* h_0 *and* h_1*.*

2.2 Heuristic methods

One commonly used assumption in color image processing is that the hue (the ratios between the color components) is smoother than the color components. Although this assumption only holds within the boundary of the objects in the image, it has been extensively used for demosaicking. Cok [33], after a reconstruction of the green channel with a bilinear interpolation, exploits the hue smoothness to reconstruct red and blue colors by interpolation of the red-to-green and blue-to-green ratios. Many other approaches follow a similar strategy, or, alternatively, find to be preferable to interpolate the difference of the color components, namely $R - G$, and $B - G$, instead of the color ratios [1]. Therefore, the reconstruction of the green channel becomes very relevant, since a good estimation of this component improves also the reconstruction of the red and blue values.

Many other approaches are based on adaptive interpolation, and the reconstruction is performed in a directional way, so the interpolation is carried out along edges rather than across them. These algorithms use as horizontal and vertical edge-classifiers the gradient [60, 80], the laplacian operator [2, 3, 55], or the jacobian [72], and then they interpolate the green image along the selected direction. Among these methods, it is interesting to analyze the strategy proposed in [3, 55]. In fact, Hamilton and Adams observe that to assume smoothness of the color differences is equivalent to state that there is a correlation between the high-frequencies of the different color components and this correlation can be exploited for the reconstruction of the green channel. They propose to improve the interpolated green image by adding a high-frequency component taken by the red and blue values, as in the scheme of Fig. 2.2(a), where the downsampled green signal G_s is interpolated with a directional low-pass filter $h_0 = [0.5,\ 1,\ 0.5]$ and the resulting output is corrected with the mid-frequency contribution selected by the filter h_1 applied to the red (or blue) component R_s (B_s), which gives the estimate

$$\hat{G}(n) = h_0 * G_s(n) + h_1 * R_s(n), \tag{2.1}$$

where n denotes the horizontal or vertical coordinate, according to the direction of interpolation. It is found that a good choice for h_1 is the FIR $[-0.25,\ \ 0,\ \ 0.5,\ \ 0,\ -0.25]$ (see Fig. 2.2(b)). Details about this technique will be given in Chapter 3.

Another efficient edge-directed technique is proposed in [31] where the edge estimation is performed by analyzing the variances of the color differences, while Tsai and Song [136] exploit the correlation between the color components to introduce a hard-decision rule and individuate the best direction for the interpolation.

Another class of algorithms [79, 89, 111] propose a weighted sum for the reconstruction of the colors, where every missing sample is estimated by its neighboring pixels and the weights are calculated on the basis of the edge directions. Some weighted sum-based techniques are considered also by Lukac *et al.* in [94, 95, 96] and many other papers, jointly to suitable assumptions on the smoothness of the color ratios or the color differences. Instead, Chang and Chen [26] present a weighting policy based on the stochastic characteristics of uniformly oriented edge indicators, modeling the image as a 2-D locally stationary Gaussian process.

Other more complex approaches are based on different techniques such as pattern matching [139], pattern recognition [33] or the use of a threshold-based variable number of gradients measured in a neighborhood of the missing pixel [25]. Instead, Freeman [46] proposes to use the median filter to improve the reconstruction near the edges, and this approach is extended in [54] considering a vector median filtering. Another non-linear technique is presented by Ramanath *et al.* in [121], exploiting the bilateral filters [134].

2.3 Demosaicking by directional interpolations

A very interesting demosaicking technique applies the scheme of Fig. 2.2(a), proposed by Hamilton and Adams, both along the horizontal and the vertical direction, producing two full color images. This leads to two estimation candidates for each pixel and the decision for the best one is made *a posteriori*. In [64] this decision is performed exploiting the local homogeneity of the image, as measured in the CIELab color space. For each pixel the direction of interpolation that gives the most homogeneous neighborhood around the estimated pixel is chosen. Then, an iterative refining step is applied, based on a median filtering of the color differences. Instead, in [140] the decision is performed observing which directional reconstruction preserves the highest correlation of the color gradients, as usually found in natural images. While the approaches proposed in [64, 140] requires two interpolations of the full color image and a decision between them involving all the three color components, in [p4] only the green channel is interpolated both horizontally and vertically and the decision for the best interpolation is carried out on the basis of the estimated values and of the CFA samples, producing an accurate estimate of the green component. Then, the missing red and blue samples are reconstructed exploiting the correlation with the green ones. This approach is described in detail in Chapter 3.

An alternative to this procedure consists in the fusion of the horizontally and the vertically interpolated values. The reconstructed image is obtained as a weighted sum of both the estimates, and the weights are computed in order to prefer the interpolation along the direction of the edges with respect to across them. This method is used in [p5, 109, 143]. The weights are computed in [143] with a linear minimum mean square-error estimation, while in [p5] some properties of the wavelet transforms are explored (this approach will be analyzed in Chapter 4).

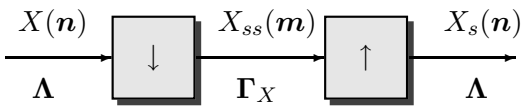


Figure 2.3: *Scheme of image sampling and zero-holder interpolation.*

Finally, Paliy *et al.* [109] use the local polynomial approximation (LPA) and the paradigm of the intersection of confidence intervals to fuse the two directional interpolations.

2.4 Frequency domain approaches

Many demosaicking approaches are based on a frequency domain analysis. Using the multidimensional notation described by Dubois in [38], an image acquired with a CFA is given by the sum of the three color components $X_s(\boldsymbol{n})$, for $X = R, G, B$, and $\boldsymbol{n} \in \boldsymbol{\Lambda}$, that are related to the full-resolution components $X(\boldsymbol{n})$ as in Fig. 2.3, where

- $\boldsymbol{\Gamma}_X$ is the sampling structure over which the color component X is acquired;
- $X_{ss}(\boldsymbol{m})$, for $\boldsymbol{m} \in \boldsymbol{\Gamma}_X$, is the sampled version of X defined on the structure $\boldsymbol{\Gamma}_X$;
- $X_s(\boldsymbol{n})$ is the zero-holder interpolated image.

The input-output relation between $X_s(\boldsymbol{n})$ and $X(\boldsymbol{n})$ can be expressed in the frequency domain as

$$X_s(\boldsymbol{\omega}) = \frac{1}{(\boldsymbol{\Lambda}:\boldsymbol{\Gamma}_X)} \sum_{\boldsymbol{k} \in \boldsymbol{\Gamma}_X^*} X(\boldsymbol{\omega} + \boldsymbol{k}), \tag{2.2}$$

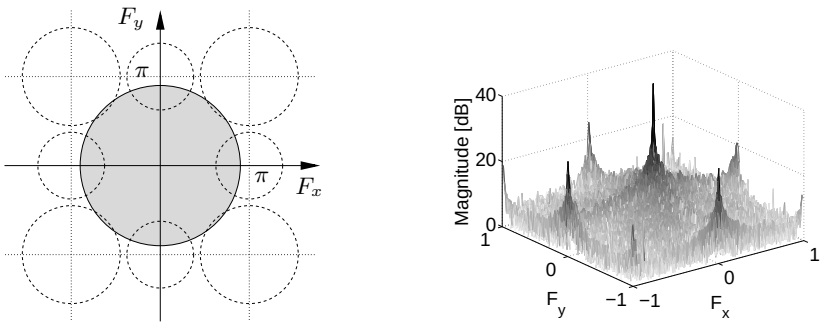
where $(\boldsymbol{\Lambda}:\boldsymbol{\Gamma}_X)$ is the index of $\boldsymbol{\Gamma}_X$ in $\boldsymbol{\Lambda}$ and $\boldsymbol{\Gamma}_X^*$ the reciprocal lattice of $\boldsymbol{\Gamma}_X$ [38].

Therefore, denoting with ω_1 and ω_2 the horizontal and the vertical frequencies, respectively, the following relations are found for images acquired with the Bayer pattern,

$$\begin{aligned} R_s(\omega_1,\omega_2) &= \frac{1}{4}[R(\omega_1,\omega_2) - R(\omega_1-\pi,\omega_2-\pi) + R(\omega_1-\pi,\omega_2) - R(\omega_1,\omega_2-\pi)] \\ G_s(\omega_1,\omega_2) &= \frac{1}{2}[G(\omega_1,\omega_2) + G(\omega_1-\pi,\omega_2-\pi)] \\ B_s(\omega_1,\omega_2) &= \frac{1}{4}[B(\omega_1,\omega_2) - B(\omega_1-\pi,\omega_2-\pi) + B(\omega_1-\pi,\omega_2) - B(\omega_1,\omega_2-\pi)]. \end{aligned} \tag{2.3}$$

In [49] it is observed that the spectrum of the green component presents less aliasing replicas with respect to the red and blue channels. Therefore it is proposed to estimate the green component with a diamond shape 2-D filter and to compute the red and green samples with a rectangular filter. Moreover, since it is assumed that the color components share the high-frequency values, the high-frequency component of the green is used to improve the reconstruction of the red and blue channels and to reduce the aliasing in the image.

Relevant considerations for the design of frequency domain demosaicking approaches come from the work of Alleysson *et al.* [6]. They observed that the spectrum of an image sampled



(a) (b)

Figure 2.4: *(a) Support of the spectrum of a CFA image: luminance component (filled circle) and chrominance components (dotted circles); (b) Spectrum of the test image lighthouse after sampling with the Bayer pattern.*

according to the Bayer pattern is given by the sum of the spectra of the three color components reported in (2.3). Then it can be expressed as (see Fig. 2.4(a))

$$\mathcal{I}_s(\omega_1,\omega_2) = L(\omega_1,\omega_2) + C_1(\omega_1-\pi,\omega_2-\pi) + C_2(\omega_1-\pi,\omega_2) - C_2(\omega_1,\omega_2-\pi), \quad (2.4)$$

where

$$L(\omega_1,\omega_2) = \frac{R(\omega_1,\omega_2)+2G(\omega_1,\omega_2)+B(\omega_1,\omega_2)}{4} \quad (2.5)$$

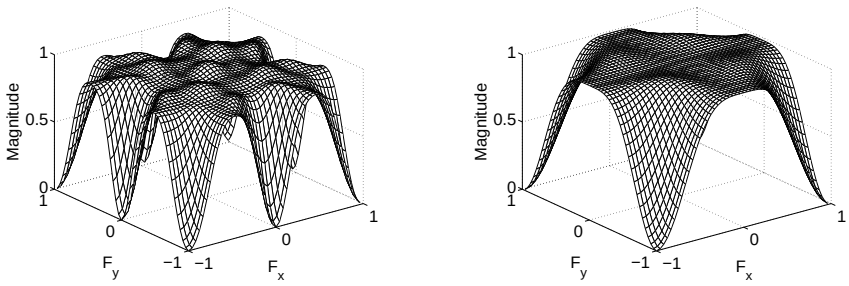
$$C_1(\omega_1,\omega_2) = \frac{-R(\omega_1,\omega_2)+2G(\omega_1,\omega_2)-B(\omega_1,\omega_2)}{4} \quad (2.6)$$

$$C_2(\omega_1,\omega_2) = \frac{-R(\omega_1,\omega_2)+B(\omega_1,\omega_2)}{4}. \quad (2.7)$$

$L(\omega_1,\omega_2)$ can be considered as a *luminance* component representing the achromatic visual information of the image, while $C_1(\omega_1,\omega_2)$ and $C_2(\omega_1,\omega_2)$ represent the *chrominances*. Therefore the CFA-sampled image is equivalent to the sum of the luminance component and the modulated chrominances. For instance, Fig. 2.4(b) shows the spectrum of the test image *lighthouse* after acquisition with the Bayer pattern. In [6] it is proposed to obtain an estimate $\hat{L}$ of the luminance with a low-pass filter removing the frequencies around $(\pm\pi,\pm\pi)$, $(0,\pm\pi)$, and $(\pm\pi,0)$ (see Fig. 2.5(a)), while the chrominances $\hat{C}_1$ and $\hat{C}_2$ can be obtained by modulating the signal $\mathcal{I}_s - \hat{L}$. Finally, the color components are computed as

$$\begin{bmatrix} \hat{R} \\ \hat{G} \\ \hat{B} \end{bmatrix} = \begin{bmatrix} 1 & -1 & -2 \\ 1 & 1 & 0 \\ 1 & -1 & 2 \end{bmatrix} \begin{bmatrix} \hat{L} \\ \hat{C}_1 \\ \hat{C}_2 \end{bmatrix}. \quad (2.8)$$

A similar strategy is exploited also in [22], where the chrominances are estimated using suitable modulations and low-pass filters, then the luminance component is reconstructed. Local adaptivity is included in the estimation by extracting the edge-information from the low-frequency region of the luminance. Instead, Dubois proposes to use complementary asymmetric filters to estimate the chrominances in [39] and introduces a least-square filter design to



(a) (b)

Figure 2.5: *(a) Frequency response of the filter to estimate the luminance proposed in [6]; (b) frequency response of the filter proposed in [87] to reconstruct the luminance in the green pixels.*

obtain an adaptive reconstruction in [40].

An interesting analysis is carried out in [87]. In fact, it is observed that, due to the data sampling structure of the CFA, two different states are observed in the spectrum of a CFA sampled image. In the red and blue positions the different frequency components are modulated as in Fig. 2.4(a), while in the quincunx-placed green pixels the color difference terms modulated at $(0, \pm\pi)$ and $(\pm\pi, 0)$ vanish. Therefore, in these locations there is less spectral overlapping between the chrominances and the luminance than in the former case, and the estimation of the luminance can be accomplished with a lower-order low-pass filter with respect to the filter proposed by Alleysson *et al.*. Then, Lian *et al.* propose to estimate the luminance component in the green pixels with the filter whose frequency response is reported in Fig. 2.5(b), while in the red and blue positions the luminance is computed with an adaptive strategy. After a full-resolution estimate of the luminance $\hat{L}$ is obtained, the color components are reconstructed through bilinear interpolation of $R - \hat{L}$, $G - \hat{L}$, and $B - \hat{L}$, respectively, exploiting the color values available from the CFA. Indeed, due to the correlation between the color components in the high-frequency region, the high-frequencies of the luminance are strongly correlated to the ones of the red, green and blue channels.

2.5 Wavelet-based methods

The correlation between the details of the different color components is exploited by some approaches in a wavelet framework. Gunturk *et al.* in [52] observe that, when a wavelet decomposition of the color values is applied using an appropriate wavelet basis, the high-frequency subbands of the color components are highly correlated, while the difference between the colors mainly resides in the coarsest resolution. They reconstruct the color image using the *Projections Onto Convex Sets* (POCS) technique, defining two constraint sets. The "observation" constraint ensures that the interpolated image is consistent with the observed data, and the "detail" constraint imposes similar high-frequency components in the color channels. These two constraint set are shown to be convex. After an edge-directed interpolation of the three channels (using the approach in [55] or similar techniques), each of them is decomposed into four subbands and the high-frequency subbands of red and blue are updated by the projection

onto the "detail" constraint (exploiting the similarity with the high-frequency subbands of the green). After the reconstruction with the synthesis filterbank, a projection onto the "observation" constraint is performed by inserting the observed data in their corresponding locations of the color components. The application of these two projections is repeated for a given number of iterations. A similar approach is proposed also by Li in [83], where a different analysis filterbank is used and a stopping criterion for the iterative stage is described.

In [37], after a simple demosaicking procedure, an estimate of the luminance component is obtained and its wavelet coefficients are used to correct the coefficients of the three color components. Instead, in [30] each of the three color channels reconstructed with an initial demosaicking algorithm is sampled, obtaining four polyphase components, namely X_{00}, X_{01}, X_{10}, and X_{11}, for $X = R$, G and B. Note that, with the pattern shown in Fig. 2.1(a), the polyphase components G_{00}, G_{11}, R_{01}, and B_{10} belong to the observed CFA image, thus their high-frequencies are not affected by interpolation errors. Successively, all the polyphase components are wavelet decomposed, and in the red positions the high-frequencies of the green and blue polyphase components are replaced with the high-frequencies of the red component. The same procedure is performed in the green and blue locations. Finally, an inverse wavelet transformation is applied to all the components and the full-resolution color channels are reconstructed.

A different approach is followed in [129], where the high frequencies of the interpolated component are refined exploiting the inter-scale correlation of the discrete wavelet transform.

2.6 Reconstruction approaches

The CFA-sampled image $\mathcal{I}_s(\boldsymbol{n})$, where $\boldsymbol{n}$ represents the pixel location, can be represented as the output of the following image formation model,

$$\mathcal{I}_s(\boldsymbol{n}) = H\mathcal{I}(\boldsymbol{n}) + \eta(\boldsymbol{n}), \tag{2.9}$$

where $\mathcal{I}(\boldsymbol{n})$ denotes the original image, $\eta(\boldsymbol{n})$ is an additive noise introduced by the sensor (uncorrelated to $\mathcal{I}(\boldsymbol{n})$), and H is a linear operator that takes into account the image acquisition process with the CFA. Therefore, demosaicking corresponds to the inverse problem of (2.9), that is to find an estimate $\hat{\mathcal{I}}(\boldsymbol{n})$ of the original image $\mathcal{I}(\boldsymbol{n})$, given the observed data $\mathcal{I}_s(\boldsymbol{n})$.

One possible solution can be achieved by the *maximum a posteriori probability* (MAP) formulation, where $\mathcal{I}_s$, $\mathcal{I}$, and η are assumed random processes and the MAP estimate is given by

$$\hat{\mathcal{I}} = \arg\max_{\mathcal{I}} p(\mathcal{I}|\mathcal{I}_s), \tag{2.10}$$

where $p(\mathcal{I}|\mathcal{I}_s)$ denotes the conditional probability density function of the random process I given $\mathcal{I}_s$. Using the Bayes rule, we obtain

$$\hat{\mathcal{I}} = \arg\max_{\mathcal{I}} p(\mathcal{I}_s|\mathcal{I})p(\mathcal{I}). \tag{2.11}$$

To find $\hat{\mathcal{I}}$ using (2.11) needs the knowledge of the probability density function (PDF) $p(\mathcal{I})$ for

the original image $\mathcal{I}$.

Brainard [16] assumes that the images lie in an appropriate space $\mathcal{B}$, which basis is composed by spatial sinusoids in each color bands. Therefore, the vector representation of $\mathcal{I}$ is given by

$$\boldsymbol{i} = \boldsymbol{B}\boldsymbol{w}, \tag{2.12}$$

where $\boldsymbol{w}$ is a multivariate gaussian process with PDF $\mathcal{N}(\boldsymbol{m}_w, \boldsymbol{K}_w)$, with $\boldsymbol{m}_w$ and $\boldsymbol{K}_w$ denoting the mean and the covariance matrix, respectively, and the columns of the matrix $\boldsymbol{B}$ are the basis of the space $\mathcal{B}$. Then, the PDF of the image $\boldsymbol{i}$ is defined as $\mathcal{N}(\boldsymbol{B}\boldsymbol{m}_w, \boldsymbol{B}\boldsymbol{K}_w\boldsymbol{B}^T)$. Due to the formulation chosen for $\mathcal{B}$, the spectra of the natural images fall off rapidly and the different color components are positively correlated. This technique is extended in [88].

Another Bayesian approaches to demosaicking is proposed in [102]. In this case the Markov random fields (MRF) model is considered for the *a priori* probability and a simulated annealing technique is applied. The MRF model is proposed also in [58] to describe both the luminance and the chrominances components, and the solution is found using POCS or the steepest descent technique. In [59] the stereable wavelets are used to decompose the image into a set of oriented subbands whose coefficients have laplacian distribution. Therefore, the *a priori* model for the image is given by the sum of the laplacian PDF of any subband.

An alternative formulation to find a solution to the inverse problem of (2.9) is given by the *minimum mean-squared error* (MMSE) method. In this case, the best linear estimate $\hat{\mathcal{I}}$ is obtained minimizing the average mean square error

$$E\left[\|\mathcal{I} - \hat{\mathcal{I}}\|^2\right]. \tag{2.13}$$

This leads to the following solution,

$$\hat{\mathcal{I}} = r_{\mathcal{I}\mathcal{I}_s}\left(r_{\mathcal{I}\mathcal{I}}\right)^{-1}\mathcal{I}_s, \tag{2.14}$$

where $r_{\mathcal{I}\mathcal{I}_s}$ denotes the correlation function between $\mathcal{I}$ and $\mathcal{I}_s$, and $r_{\mathcal{I}\mathcal{I}}$ the autocorrelation of $\mathcal{I}$. For gaussian distributions the MAP and MMSE estimates are identical.

A demosacking technique based on a MMSE approach is proposed in [131] with a frequency domain analysis and a scale-invariant prior for the original image, that is assumed to have the following power spectral density,

$$\mathcal{P}_{\mathcal{I}}(\omega_r, \omega_\theta) = \mathcal{F}_{\omega_r,\omega_\theta}\left(E[\mathcal{I}\mathcal{I}^*]\right) = \frac{1}{\omega_r^2} f(\omega_\theta)\Gamma_0, \tag{2.15}$$

where ω_r and ω_θ are the radial and angular frequencies, respectively. The angular distribution $f(\omega_\theta)$ is usually taken as constant, and Γ_0 is another constant arising from surface spectral reflectances. This image prior is used to derive an optimal constrained MMSE estimator. A proof for the existence of the constrained solution to (2.13) is also given, providing key insights required for the construction of the linear operator.

A MMSE approach is considered also in [135], where Trussell and Hartwig exploit simple assumptions on color images to propose a cost-effective method reducing the dimensions of the huge matrices and allowing fast implementations. Instead, Portilla *et al.* in [118] introduce

a joint spatial-chromatic correlation model trained on a small set of images. A similar idea to obtain the prior model for the color images is presented also in [23].

An alternative to find a solution to the inverse problem of (2.9) is given by the regularization methods. As for the statistical approaches, the role of regularization is to exploit the *a priori* knowledge about natural images, in order to convert an ill-posed problem into a well-posed one [34, 74]. Regularization is widely used in many aspects of reconstruction and restoration of digital images, such as denoising, deblurring, motion estimation, magnification or super-resolution.

The *regularized solution* $\hat{\mathcal{I}}$ is defined as the solution of the following problem,

$$\hat{\mathcal{I}} = \arg\min_{\mathcal{I}} \left\{ \Psi(\mathcal{I}, \mathcal{I}_s) + \sum_k \lambda_k J_k(\mathcal{I}) \right\}, \tag{2.16}$$

where the first term $\Psi(\mathcal{I}, \mathcal{I}_s)$, called *data-fidelity term*, denotes a measure of the distance between the estimated image and the observed data, while the terms $J_k(\mathcal{I})$ are some regularizing constraints based on a *a priori* knowledge of the original image. The regularization parameters λ_k control the tradeoff between the different terms. The theory about regularization will be deeply analyzed in Chapter 5.

Keren and Osadchy propose a regularization approach to demosaicking in [77], exploiting two regularizing constraints. The first one is composed by the first-order derivatives in the horizontal, vertical and diagonal directions, respectively, and is used to impose smoothness on each color component. The second constraint is given by the sum of the outer product of each pixel vector $[R(\boldsymbol{n}), G(\boldsymbol{n}), B(\boldsymbol{n})]$ with the vectors representing its neighboring pixels. In this way the regularization forces also the color components to be correlated. This technique is considered also in the joint demosaicking and super-resolution approach presented in [43] that will be analyzed in Section 2.8. Another regularization method to demosaicking is proposed in [126], using the total-variation criterion to impose smoothness both on the single color components and on the color-differences. Other approaches based on the regularization methods are presented in [p6, p8] and will be described in Chapter 5.

Another competitive demosaicking algorithm is proposed in [103], exploiting the optimal recovery estimation theory. It estimates the missing pixels of some local image patches from the known pixels, under the assumption that each local patch belongs to a known quadratic signal class. The optimal recovery estimate for the missing value minimizes the maximum error over all possible vectors belonging to the signal class that have the same known samples.

The sparse nature of the color image is exploited also in [97], where a demosaicking approach is proposed in the spirit of the compressed sensing theory. Natural images are assumed to admit a sparse decomposition over a redundant dictionary. Therefore, a well-adapted dictionary is designed and learned with the K-SVD algorithm, in order to be used for the reconstruction of the CFA-sampled images.

2.7 Joint demosaicking and zooming

Many digital devices, after the demosaicking procedure, need a zooming of the image to increase the resolution. Many techniques to image interpolation have been proposed (see Chapter 7), and usually zooming and demosaicking are considered separately. However, this solution presents some relevant drawbacks:

- the computational cost is high due to the application of both demosaicking and zooming. Moreover the zooming algorithm has to be applied to each color component;
- the color artifacts introduced by the demosaicking reconstruction are expanded by the zooming;
- some image interpolation techniques are adaptive, and the detection of the edges can be affected by the demosaicking artifacts, especially in correspondence of details.

Therefore, in literature, some joint demosaicking and zooming approaches are considered. In this way, the raw data are processed according to the desired resolution, only one edge-detection analysis is performed instead of applying it both in the demosaicking and in the zooming algorithm, and the analogies between these two interpolation procedures are exploited. Moreover, the computational cost is kept lower with respect to the application of both demosaicking and zooming.

Recently two approaches to the joint demosaicking and zooming problem have been proposed in [32, 145]. Both techniques start with the reconstruction of the green component and produce an estimate of this component at the same resolution of the CFA image, exploiting two directional interpolations and a decision for the best reconstruction, as explained in Section 2.3. Then, the green component is magnified at the desired resolution and is used to drive the interpolation of the red and blue channels exploiting the correlation between the color values. Both approaches give better performances than to apply demosaicking and zooming separately.

Another method to increase the resolution starting from the raw data is proposed in [90, 91]. In this case, an estimation of the CFA-sampled image at the desired resolution is performed, then classical demosaicking approaches can be applied.

2.8 Joint demosaicking and super-resolution

Another approach to enhance the spatial resolution is to fuse several low-resolution images of the same scene, in order to produce a high-resolution image (or a sequence of them). It is called *super-resolution* and is proved to be useful in many practical situations where multiple frames of the same scene are available, including medical imaging, satellite imaging, videosurveillance and other video applications. Many techniques to super-resolution have been proposed and an overview of them can be found in [110]. Since the low-resolution images are often acquired by digital cameras provided with a CFA, it is interesting to perform super-resolution from the raw-data directly, that is to consider demosaicking and super-resolution jointly.

A solution to this problem is proposed in [43] with a regularization method that exploits several regularizing constraints. One of them imposes smoothness to the luminance component

with the bilateral total-variation technique, an extension of the total-variation criterion (see Chapter 5) in the spirit of the bilateral filters [41, 134]. A quadratic penalty term is used to describe the bandlimited characteristic of the chrominances, and another term penalizes the mismatch between locations or orientation of edges across the color bands, generalizing the vector product-based term introduced in [77]. The data fidelity term measuring the similarity between the resulting high-resolution image and the original low-resolution images is based on the ℓ_1 norm. A steepest descent optimization is applied to minimize the cost function. Another regularization approach to joint demosaicking and super-resolution is described in [51], using the Tikhonov method of regularization.

A different strategy is followed in [137]. The luminance and the chrominances components are extracted from the CFA image according to the approach of Alleysson *et al.* described in Section 2.4, then are interpolated to obtain their high-resolution versions. Finally, the high-resolution color image is reconstructed.

2.9 Joint demosaicking and denoising

The acquisition of an image introduces a noisy component due to the photon counting process and the electrical and thermal effects in the sensor. Therefore, a digital camera pipeline has to consider also a *denoising* procedure to remove the noise from the acquired image. In literature, many denoising approaches have been proposed, exploiting, for instance, the Wiener filtering, the total variation regularization [124] or the bilateral filters [134]. Most of the denoising approaches apply linear transformations that are able to separate low and high frequencies (such as the wavelets, the curvelets or the contourlets). In fact, it is observed that natural images concentrate the energy in the lowest frequencies, while in the high-frequency subbands the energy is localized only in correspondence of the details of the image. On the other hand, if the noise is assumed white, its distribution is constant over all the spectrum. Therefore, many denoising approaches [27, 35, 114, 119] apply a linear transformation (such as the wavelets), then "clean" the high-frequency subbands simply thresholding them. Finally, an inverse transformation of the "cleaned" coefficients is done to obtain the denoised image. This procedure is based on the idea that large coefficients, which are kept, belong to the details of the image, whereas the noise is distributed across small coefficients, which are canceled.

Denoising can be performed before demosaicking [73], or after the color reconstruction. However, both the solutions present relevant drawbacks. In the first case denoising has to be accomplished considering separately the three color components, and it does not allow for analyzing the image at the maximum resolution, which is important to exploit the frequency sparsity of the natural images. On the other hand, if demosaicking is performed before denoising, it has to be applied on a noisy image, making difficult the edge estimation that is an important task of the adaptive approaches. Moreover, the demosaicking procedure modifies the statistical characteristics of the noise, and it complicates the application of the denoising approaches that are based on particular assumptions about the noise. It has to be pointed out also that to denoise color images requires to remove the noise on all the three color components, thus increasing the computational cost.

Therefore a more efficient solution is to consider denoising and demosaicking jointly. This approach is proposed in [65] using the total least square denoising to remove both signal-independent and signal-dependent noise. Another technique is described in [62] where a Bayesian hierarchical modeling is used to capture the second-order statistics in the wavelet domain. An EM (*Expectation-Maximization*) algorithm is coupled with the Bayesian model, in order to obtain the reconstruction of the unobserved data and the denoising of the whole image starting from the available data. Instead, in [63] a wavelet transform of the noisy CFA-image is applied, and the wavelet coefficients of the luminance and the chrominances components are estimated and denoised. Finally, an inverse wavelet transform is applied and the color image is reconstructed.

Zhang *et al.* in [144] recover the green component by estimating the color differences with a MMSE approach that exploits both spectral and spatial correlations to simultaneously suppress sensor noise and interpolation error. Then, the CFA channel-dependent noise is removed from the reconstructed green channel with a wavelet-based approach. Finally, also the red and blue channels are estimated and denoised. Instead the method in [108] is based on the local polynomial approximation (LPA) and the paradigm of the intersection of confidence intervals to design and choose suitable filters to denoise and interpolate the CFA samples.

Another joint demosaicking and denoising approach will be proposed in Chapter 6.

2.10 Quality assessment of demosaicking approaches

In literature, a commonly used approach for evaluating demosaicking algorithms consists in choosing some full-color test images, sampling them according to the Bayer pattern (or other CFA arrangements) and estimating them with a demosaicking procedure. Lower is the difference between the original images and their respective reconstructed versions, better is the effectiveness of the algorithm. Usually the *Mean Square Error* (MSE) is considered, that, for each color component $X(n_1, n_2)$, with $X = R, G, B$, and $n_1 = 1, \dots N_1$, $n_2 = 1, \dots N_2$, is given by

$$\text{MSE}(X) = \frac{1}{N_1 N_2} \sum_{n_1=1}^{N_1} \sum_{n_2=1}^{N_2} \left(\hat{X}(n_1, n_2) - X(n_1, n_2) \right)^2 . \tag{2.17}$$

Alternatively a logarithmic measure is preferred, such as the *peak signal-to-noise ratio* (PSNR) or the *color peak signal-to-noise ratio* (CPSNR), defined as [6]

$$\text{CPSNR} = 10 \log_{10} \frac{255^2}{\frac{1}{3} \sum_{X=R,G,B} \text{MSE}(X)}. \tag{2.18}$$

In order to give an evaluation closer to the human visual perception with respect to the CPSNR measure, often the S-CIELAB metric [146] is used to quantify the performance of the different demosaicking algorithms. S-CIELAB is an extension of the CIE$L^*a^*b^*$ ΔE color difference formula exploiting also a spatial pre-processing step that incorporates the pattern-color sensitivity measurements proposed by Poirson and Wandell [116, 117]. Moreover, often these numerical considerations are combined with a visual inspection of some detailed regions where many demosaicking methods fail.

Test-images are commonly taken from the Kodak dataset, that is available in the website `http://www.cipr.rpi.edu/resource/stills/kodak.html`. These images are film captured and then digitized at the resolution of 512×768. In this dataset there are a lot of details that allow for an immediate evaluation of the quality of the demosaicking procedures. However, despite its popularity in literature, the film captured images of the Kodak dataset are not very representative of the data acquired by the most recent digital cameras [84]. Therefore, often it is preferable to test the demosaicking approaches using raw data captured using real digital cameras. In this case, however, knowledge about the "original" image is not available, then it is not possible to measure the quality with quantitative metrics such as MSE, CPSNR or S-CIELAB. A study about a perceptual assessment of the quality of some demosaicking algorithms is reported in [88]. In the following chapters we will provide some comparisons between the proposed approaches and the techniques presented in the literature considering both numerical evaluations over the Kodak dataset and visual inspections with raw images captured with digital cameras.

Chapter 3

Demosaicking with directional filtering and *a posteriori* decision

The first part of this chapter describes the demosaicking approach recently proposed in [p4]. It is based on two directional interpolations of the green component and a a posteriori decision for the best reconstruction. Then, also the red and blue channels are estimated. A refining step is included to further improve the resulting reconstructed image. The described approach requires a limited computational cost and gives good performance even when compared to more demanding techniques. Therefore, this algorithm candidates itself for implementation in simple low-cost cameras or in video capture devices with high values of resolution and frame-rate.

The second part of the chapter presents a technique to reduce the artifacts introduced by directional demosaicking algorithms, proposed in [p2]. It can be applied directly after the color interpolation, or as an off-line post-processing.

3.1 Introduction

One of the most promising demosacking technique is based on directional filtering (see Section 2.3). In particular, the algorithms proposed in [64, 140] initially compute two estimates f^H and f^V of the full color image. The green components G^H and G^V of these images are obtained through horizontal and vertical interpolation, respectively, then the red and blue components are reconstructed using a bilinear interpolation of the color differences $R - G^H$ and $B - G^H$ for the horizontally estimated image f^H, and of $R - G^V$ and $B - G^V$ for the vertically estimated image f^V. Next, for each pixel a choice between f^H and f^V is performed. This approach proved to give good performance, because is able to detect the edge directions and avoid interpolation across the edges. In this way the introduction of color artifacts is limited and the resulting images have both good visual quality and low MSE with respect to the original ones [52]. However this approach requires to compute and compare two full color images.

In this chapter we describe a more effective approach to directional interpolation. The decision of the most suitable direction of interpolation is made on the basis of the reconstructed green component only. Once the choice is made, the red and blue components are interpo-

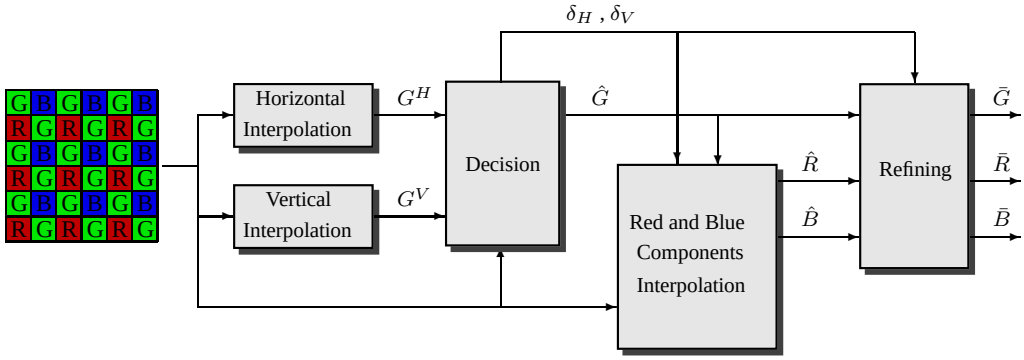


Figure 3.1: *The complete scheme of the proposed algorithm.*

lated. In this way the two directional interpolations and the decision concern only one color component and not all the three channels. Moreover this approach requires the decision only in a half of the pixels of the image, precisely where the sensor did not capture the green samples. Furthermore, since in this case the estimate of the green component after the decision is more accurate, a more efficient reconstruction of red and blue is possible. In this way the computational cost of the overall algorithm is reduced and the performances are improved.

The next section describes the proposed algorithm in detail. In Section 3.3 an additional refining step to improve the quality of the reconstructed image is presented. The computational cost and the performance of the proposed approach are discussed in Section 3.4 and compared to those of other recent demosaicking techniques.

Moreover in Section 3.5 a technique to reduce the artifacts introduced by directional algorithms is described. It can be useful to improve the performances of demosaicking approaches such as those presented in the next section or the methods proposed in [64, 140].

3.2 Proposed algorithm

Figure 3.1 shows the complete scheme of the proposed demosaicking technique, and in the following we will analyze its elements in detail.

3.2.1 Directional green interpolation

The first step of the algorithm is to reconstruct the green image along horizontal and vertical directions. To interpolate the Bayer samples, we apply a 5 coefficients FIR filter. We do not use a longer filter because this would produce zipper effect near the edges. The filter we used is the same as [3], [64] and [140], and we rederived its coefficients on the basis of the following considerations.

We can note that, along a row (or a column) of the Bayer pattern, the green signal is subsampled with a factor 2. In the frequency domain this gives:

$$G_s(\omega) = \frac{1}{2}G(\omega) + \frac{1}{2}G(\omega - \pi), \tag{3.1}$$

where $G(\omega)$ and $G_s(\omega)$ denote the Fourier transform of the original green signal and of the

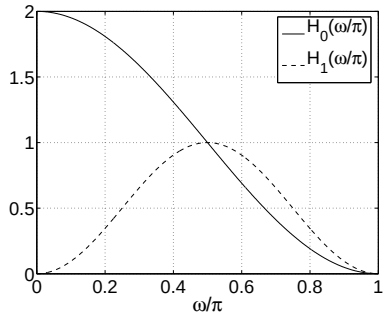


Figure 3.2: *Frequency response of the two filters h_0 and h_1.*

down-sampled signal, respectively. Therefore, if $G(\omega)$ is band-limited to $|\omega| < \pi/2$, the ideal interpolation filter to perform the reconstruction would be

$$H_{id}(\omega) = 2 \text{ rect}\left(\frac{\omega}{\pi}\right), \tag{3.2}$$

since it eliminates the aliasing component $1/2\ G(\omega - \pi)$. The only FIR filter with 3 coefficients that we can apply to $G_s(\omega)$ without modifying the average value of the samples is $h_0 = [0.5,\ \ 1,\ \ 0.5]$ (see Fig. 3.2).

However, this non-ideal filter does not remove totally the aliasing. In fact, after filtering we have

$$\hat{G}(\omega) = G_s(\omega)H_0(\omega) = \frac{1}{2}G(\omega)H_0(\omega) + \frac{1}{2}G(\omega - \pi)H_0(\omega), \tag{3.3}$$

where the second term denotes the aliasing component.

A solution to decrease the aliasing effect and to improve the mid-frequency response could be to take advantage of the information coming from the high-bands of the red and blue signals, since it is well-known that the high-frequencies of the color channels are highly correlated.

In a green-red row, the red component is sampled with an offset of 1 sample with respect to the green signal. Therefore its Fourier transform results

$$R_s(\omega) = \frac{1}{2}R(\omega) - \frac{1}{2}R(\omega - \pi), \tag{3.4}$$

where $R(\omega)$ is the Fourier transform of the original red signal. If we interpolate it with a filter h_1 and we add the resulting signal to (3.3) as in Fig. 3.3, we have

$$\begin{aligned} \hat{G}(\omega) &= \frac{1}{2}G(\omega)H_0(\omega) + \frac{1}{2}G(\omega - \pi)H_0(\omega) + \\ &+ \frac{1}{2}R(\omega)H_1(\omega) - \frac{1}{2}R(\omega - \pi)H_1(\omega). \end{aligned} \tag{3.5}$$

Reminding that $R(\omega) - G(\omega)$ is slowly varying [3, 52], if h_1 is designed such that $H_1(\omega) \simeq 0$ at low-frequencies and $H_1(\omega) \simeq H_0(\omega)$ at high-frequencies, we have

$$R(\omega)H_1(\omega) \simeq G(\omega)H_1(\omega), \tag{3.6}$$

$$G(\omega - \pi)H_0(\omega) \simeq R(\omega - \pi)H_1(\omega) \tag{3.7}$$

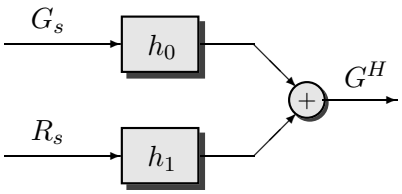


Figure 3.3: *Horizontal reconstruction of the green signal in a green-red row.*

and (3.5) could be approximated as

$$\hat{G}(\omega) \simeq \frac{1}{2}G(\omega)H_0(\omega) + \frac{1}{2}R(\omega)H_1(\omega). \tag{3.8}$$

A good choice for a filter h_1 that respects the constraints (3.6) and (3.7) is the 5-coefficients FIR $[-0.25, \quad 0, \quad 0.5, \quad 0, \quad -0.25]$ (see Fig. 3.2).

Therefore, in the following row of the Bayer-sampled image,

$$\cdots \;\; R_{-2} \;\; G_{-1} \;\; R_0 \;\; G_1 \;\; R_2 \;\; \cdots,$$

the missing green sample G_0 is estimated as

$$\hat{G}_0 = \frac{1}{2}\left(G_1 + G_{-1}\right) + \frac{1}{4}\left(2\,R_0 - R_2 - R_{-2}\right). \tag{3.9}$$

As described in Chapter 2, Adams [3] follows a similar approach to derive this filter, while in [64] Hirakawa and Parks obtain the same filter starting from different constraints and solving an optimization problem to choose the filter's coefficients. An interesting interpretation of (3.9) is supplied by Wu and Zhang in [140], where they note that (3.9) can be written as

$$\hat{G}_0 = R_0 + \frac{1}{2}\left(G_1 - \frac{R_0 + R_2}{2} + G_{-1} - \frac{R_0 + R_{-2}}{2}\right). \tag{3.10}$$

That is, this reconstruction can also be considered as a bilinear interpolation of the $R - G$ difference, where the unknown values R_1 and R_{-1} are estimated as $(R_0 + R_2)/2$ and $(R_0 + R_{-2})/2$, respectively.

The interpolation of the green values in the blue-green rows and the interpolation along the columns follows the same approach.

3.2.2 Decision

Once the green component has been interpolated along both horizontal and vertical directions and two green images have been produced, a decision has to be made to select the filtering direction that gives the best performance.

As seen in the previous sections, a natural property of the images is the smoothness of the color differences. In fact the color differences vary slowly and present abrupt changes only across the edges. Therefore, an image typically has greater values of the gradients of the color differences across the edges than along them. This property can be used to locate the presence and the direction of edges in a natural image and hence can be applied in the decision step

to detect which is the most appropriate direction of interpolation, similarly to the procedure applied in [140].

Let G^H and G^V be the two interpolated green images. For each image in every red or blue location (i.e. where the sensor acquired the red or blue values) we calculate the chrominance values $R - G^H$ (or $R - G^V$) in a red pixel, and $B - G^H$ (or $B - G^V$) in a blue pixel; namely

$$C_H(n_1, n_2) = \begin{cases} R(n_1, n_2) - G^H(n_1, n_2) & \text{if } (n_1, n_2) \text{ is a red location} \\ B(n_1, n_2) - G^H(n_1, n_2) & \text{if } (n_1, n_2) \text{ is a blue location,} \end{cases} \quad (3.11)$$

$$C_V(n_1, n_2) = \begin{cases} R(n_1, n_2) - G^V(n_1, n_2) & \text{if } (n_1, n_2) \text{ is a red location} \\ B(n_1, n_2) - G^V(n_1, n_2) & \text{if } (n_1, n_2) \text{ is a blue location,} \end{cases} \quad (3.12)$$

where n_1 and n_2 indicate the row and the column of the pixel (n_1, n_2), $1 \leq n_1 \leq M$, $1 \leq n_2 \leq N$ (M and N denote the height and the width of the image, respectively). Note that C_H and C_V are not defined in the green pixels. Next, we calculate the gradients of the chrominances and, precisely, the horizontal gradient for C_H and the vertical one for C_V,

$$D_H(n_1, n_2) = |C_H(n_1, n_2) - C_H(n_1, n_2 + 2)|, \quad (3.13)$$

$$D_V(n_1, n_2) = |C_V(n_1, n_2) - C_V(n_1 + 2, n_2)|. \quad (3.14)$$

Note that, due to the structure of the Bayer pattern, the gradients are always computed as a difference of the same kind of chrominance ($R - G$ or $B - G$). For each red or blue pixel, we then define the classifiers $\delta_H(n_1, n_2)$ and $\delta_V(n_1, n_2)$ as the sum of the gradients D_H and D_V belonging to a sufficiently large neighborhood of (n_1, n_2) (for example a 5×5 window gives good classifiers).

With a square window, both the classifiers are computed considering the same number of gradients based on the red chrominances and the same number of gradients based on the blue chrominances.

It this way the two classifiers $\delta_H(n_1, n_2)$ and $\delta_V(n_1, n_2)$ give an estimate of the local variation of the color differences along the horizontal and vertical directions, respectively, and they can be used to estimate the direction of the edges. For example, if the value of δ_H is lower than δ_V, it is likely that there is a horizontal edge instead of a vertical one.

For all the red and blue pixels we estimate the green values using the following criterion:

```
if  δ_V(n1, n2) < δ_H(n1, n2)
then
  Ĝ(n1, n2) = G^V(n1, n2);
else
  Ĝ(n1, n2) = G^H(n1, n2).
```

So, considering also the known green samples, a full resolution green image $\hat{G}$ is estimated.

An additional improvement can be included in this procedure. Usually, in natural images, the majority of the edges and the details presents cardinal orientations. Therefore, if the pixel (n_1, n_2) is placed in a detailed area, during the estimation of the green values $\hat{G}(n_1, n_2)$, it can

Locations	Bilinear Int. of $R-G$	Edge Directed [55]	Weighted Sum [79]	Proposed Algorithm
Green pixels	7.39	7.39	6.01	7.39
Blue pixels	17.03	17.52	16.88	13.39

Table 3.1: *Average MSE for the interpolation of R in the green and blue locations for the five test-images lighthouse, sail, boat, statue and window of the Kodak dataset.*

result preferable to give more weight to the gradients D_H and D_V of the pixels in the same row and column of (n_1, n_2). This can be accomplished by weighting these gradients two or three times more than the other gradients when we calculate $\delta_H(n_1, n_2)$ and $\delta_V(n_1, n_2)$. In our implementation a weight of 3 is used to this purpose.

3.2.3 Red and blue interpolation

After the green channel has been reconstructed, we have to interpolate the red and blue components. Besides the Bayer data, now we have a full resolution green image component $\hat{G}$ and the classifiers δ_H and δ_V that can be used for the reconstruction of the other two components.

In literature [1, 55, 64, 79, 80, 89, 103, 140] the most common approach for red and blue estimation consists in interpolation of the color differences $R-G$ and $B-G$, instead of R and B directly. Some particular reconstruction methods based upon the color differences have been explored, such as the weighted sum proposed in [79, 89] and [103] or the Optimal Recovery also explained in [103]. Nevertheless, the most used technique remains the bilinear interpolation, sometimes with a small modification for the reconstruction of red (or blue) component in the blue (red) pixels, where an edge-directed interpolation can be applied to interpolate the color differences along one of the two diagonal directions, selected with a laplacian operator [55].

Table 3.1 reports the average mean square reconstruction error (MSE) for the red component of five test-images using different demosaicking approaches, in the green and the blue pixels, respectively. We note that, for all the approaches, the error is higher in the blue pixels then in the green ones. Similar results are found for the reconstruction of the blue channel.

For this reason, in the green locations we apply the bilinear interpolation of the color differences, as in other techniques, since it gives good performance with a low computational cost, while we propose a different approach to reconstruct the red values in the blue pixels and the blue values in the red ones.

Referring to the estimation of the red component (the same strategy is applied for the blue one), once all the green positions are interpolated, we use these estimated samples to reconstruct red also in the blue pixels. In fact, now each blue position has four estimated red neighbors placed in the cardinal directions. Therefore, we choose to perform an edge-directed interpolation using the estimated red samples in the green location. To decide the best direction we do not need a new edge-detection but we can use the classifiers δ_H and δ_V already applied in the reconstruction of the green component (remind that δ_H and δ_V have been computed for all the red and blue pixels). As opposed to other techniques, the cardinal directions are preferred

with respect to the diagonal ones. This is justified by the fact that in natural images there are more cardinal edges with respect to the edges in other directions.

Moreover, in natural images it is a fact that red and blue components are less correlated than red and green or blue and green. For example, for the same five test images used in Table 3.1 we found an average correlation value of 0.84 between R and B, while the average correlations between G and R or B are 0.94 and 0.93, respectively. However this fact is not indicative of which color difference ($R-B$ or $R-G$) is preferable to use to obtain the estimate of the red values in the blue pixels of the Bayer pattern (the reconstruction of the blue in the red locations is totally symmetrical, hence we refer only to the interpolation of R).

In fact, the formula applied for reconstructing a red sample in a blue location is

$$\hat{R}(n) = C(n) + \frac{1}{2}\left(\hat{R}(n-1) - C(n-1) + \hat{R}(n+1) - C(n+1)\right), \tag{3.15}$$

where n denotes the horizontal or vertical coordinate, according to the direction of interpolation, and $C = \hat{G}$ if we interpolate the difference $R-G$, while $C = \hat{B}$ if we interpolate $R-B$. This equation can be rewritten as

$$\hat{R}(n) = \frac{1}{2}\left(\hat{R}(n-1) + \hat{R}(n+1)\right) - \frac{1}{2}C(n-1) + C(n) - \frac{1}{2}C(n+1), \tag{3.16}$$

that is, the reconstruction of R consists in a bilinear interpolation of the two neighboring samples, corrected with the high-frequency content of the C component selected by the filter $[-0.5,\ \ 1,\ -0.5]$.

Therefore for the correction it is preferable to use the C color most correlated to the original red samples in the high-frequencies. To select it, we calculate the high-frequency correlation coefficient in the blue locations of the Bayer pattern between the original red samples (that we have to estimate) and the available estimates of the green and blue components, namely $\rho_{R\hat{G}}$ and $\rho_{R\hat{B}}$. Two methods have been performed to compute the high-frequencies, using two different high-pass filters. The first one has coefficients $[-0.5,\ \ 1,\ -0.5]$ and it is applied along the directions indicated by the edge-detection, as in Section 3.2.3. The second is the 2-D high-pass filter

$$G = \begin{bmatrix} 0 & -0.25 & 0 \\ -0.25 & 1 & -0.25 \\ 0 & -0.25 & 0 \end{bmatrix}. \tag{3.17}$$

The results are reported in Table 3.2. We see that, in the blue locations, the high-frequencies of the estimated blue component are more correlated to the red values than to the high-frequencies of the estimated green component. Therefore the bilinear interpolation of $R-B$ gives a better approximation of the red samples than interpolation of $R-G$. This is not totally surprising, because, although the green channel is more accurately sampled in the Bayer pattern than the blue channel, here we are referring to interpolation in the locations where the sensor captured the blue component.

Therefore, in a blue position the red is estimated as follows:

`if` $\delta_V(n_1, n_2) < \delta_H(n_1, n_2)$

Test-image	with 1-D filter		with 2-D filter	
	$\rho_{R\hat{B}}$	$\rho_{R\hat{G}}$	$\rho_{R\hat{B}}$	$\rho_{R\hat{G}}$
lighthouse	0.89	0.83	0.94	0.92
sail	0.86	0.80	0.92	0.90
boat	0.92	0.85	0.96	0.93
statue	0.88	0.84	0.92	0.88
window	0.88	0.85	0.92	0.90
Average	**0.89**	**0.83**	**0.93**	**0.91**

Table 3.2: *High-frequency inter-channel correlation between the original red samples and the estimated green and blue samples, respectively, in the blue locations.*

```
then
```
$$\hat{R}(n_1,n_2) = B(n_1,n_2)+ \\ +\frac{1}{2}\left(\hat{R}(n_1-1,n_2)-\hat{B}(n_1-1,n_2)+\hat{R}(n_1+1,n_2)-\hat{B}(n_1+1,n_2)\right);$$
```
else
```
$$\hat{R}(n_1,n_2) = B(n_1,n_2)+ \\ +\frac{1}{2}\left(\hat{R}(n_1,n_2-1)-\hat{B}(n_1,n_2-1)+\hat{R}(n_1,n_2+1)-\hat{B}(n_1,n_2+1)\right).$$

For the interpolation of the blue values in red pixels, the same strategy is applied.

In Table 3.3 the performance of various techniques to reconstruct red and blue channels for several test-images are reported (the green image has been interpolated following the approach explained in Sections 3.2.1 and 3.2.2). Bilinear interpolation of the color differences, the edge-directed technique with estimation in the diagonal directions as in [55], the weighted sum of the color differences (using the weights proposed in [79]) and the proposed method are compared. Numbers in boldface indicate the smallest values. Note that the proposed algorithm outperforms other methods in many images and its performance is at least comparable to that of more complex and computationally demanding approaches. Moreover, its complexity is as low as a simple bilinear interpolation.

3.3 Refining step

The technique explained in Section 3.2 reconstructs in a fast way the full resolution image avoiding visible and annoying artifacts. However, even with an accurate selection of the edge directions, the reconstructed image may contain several errors due to the *interpolation artifacts* [64], less noticeable than misguidance artifacts (introduced by a wrong edge-estimation) but still annoying. In the proposed algorithm they can be introduced by the approximations made in the filter design (see Section 3.2.1) and, furthermore, by the low-pass characteristic of the filters used to interpolate the green component and the color differences $R-G$ and $B-G$. Note that these artifacts mainly affect the regions with high-frequency contents.

We propose to correct them by using the high-bands inter-channel correlation of the three primary colors. A good solution may consist in separating low- and high-frequency components in each pixel and replacing the high-frequencies of the unknown components with the

Test-image		Bilinear	Edge Directed [55]	Weighted Sum [79]	Proposed Algorithm
lighthouse	R	10.92	11.20	9.90	**9.70**
	B	9.19	9.54	8.90	**7.68**
sail	R	5.70	5.73	**5.05**	5.22
	B	4.30	4.46	4.79	**3.66**
boat	R	11.06	11.43	10.16	**9.24**
	B	14.33	14.40	12.52	**12.15**
statue	R	6.90	6.90	**6.16**	6.23
	B	8.74	8.82	8.07	**7.77**
window	R	5.17	5.10	4.82	**4.81**
	B	6.59	6.48	**6.02**	6.13
Average	R	7.95	8.07	7.22	**7.04**
	B	8.63	8.74	8.45	**7.48**

Table 3.3: *MSE comparison of different techniques for interpolating the red and blue channels.*

high-frequencies of the Bayer-known component. The low-frequency component is preserved unchanged since the low-frequency components of the color channels are less correlated.

For example, for a green pixel in the location (n_1, n_2), the green value can be decomposed as:

$$G(n_1, n_2) = G^{\ell}(n_1, n_2) + G^{h}(n_1, n_2), \tag{3.18}$$

where G^{ℓ} and G^{h} denote the low- and high-frequency components, respectively, and the red and blue values can be corrected replacing R^{h} and B^{h} with G^{h}. That is

$$R(n_1, n_2) = R^{\ell}(n_1, n_2) + G^{h}(n_1, n_2) \tag{3.19}$$

$$B(n_1, n_2) = B^{\ell}(n_1, n_2) + G^{h}(n_1, n_2). \tag{3.20}$$

The correction in the red and blue pixels is carried out in a similar way.

The selection of the low-frequency components is performed using a low-pass filter while the high-frequencies are calculated subtracting the low-frequency values (see Fig. 3.4). The design of this low-pass filter is very important for the performance of the refining step and has to consider the following points.

A first important issue consists in exploiting the knowledge of the Bayer data, since we are sure that they are not affected by interpolation errors. So, it results preferable that the red (blue) component in the green locations, having only two neighbors belonging to the Bayer pattern, are corrected using a 1-D low-pass filter selecting only the red (blue) positions.

For the correction of the green channel and of the red and blue colors in the blue and red pixels, one possible choice is to span all the neighborhood, for example with a 2-D filter with a 3×3 kernel. A similar approach has been recently presented and analyzed in [83], where the color differences are filtered and successively used to correct the high-frequencies of the image. However, an isotropic filtering may introduce zipper effect near the edges degrading the quality of the image since it performs the interpolation of the color differences also across the edges. A more effective approach is to select the low- and high-frequencies using a 1-D

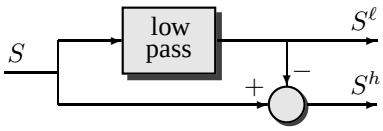


Figure 3.4: *Selection of the low-frequency and high-frequency components in the refining step ($S = R, G, B$).*

filter, so the interpolation is carried out only along the edges of the image.

To summarize, the refining step is performed as follows:

1. **Updating of the green component**. For each red location (n_1, n_2), the green and red channels are filtered with a low-pass filter along the direction selected using δ_H and δ_V. The four components $G^\ell(n_1, n_2)$, $G^h(n_1, n_2)$, $R^\ell(n_1, n_2)$ and $R^h(n_1, n_2)$ are obtained. Then, the green high-frequency values $G^h(n_1, n_2)$ are replaced with $R^h(n_1, n_2)$ and the green samples $G(n_1, n_2)$ are reconstructed. The same update is carried out for the green values in the blue locations.

2. **Updating of the red and blue components in the green locations**. For each green position, the green and the red subband values are obtained through horizontal or vertical filtering, depending on where the neighbor red values in the Bayer pattern are placed. Then, the high-frequency component of the red channel is updated with the green one and the red values are reconstructed. The update of the blue component is carried out in the same way.

3. **Updating of the red (blue) component in the blue (red) locations**. The red and blue channels are decomposed into low- and high-frequency components according to the most appropriate direction given by the comparison of δ_H and δ_V. The updated values in the neighboring pixels are used in order to obtain a more reliable estimate. Then, the red high-frequency component $R^h(n_1, n_2)$ is replaced with $B^h(n_1, n_2)$. The blue values in the red pixels are refined in a similar way.

Table 3.4 reports the mean square reconstruction error before and after the refining step for the test images considered in Section 3.2. The low-pass filter applied is the FIR $[1/3, 1/3, 1/3]$. We can notice that the updating improves the quality of the images, reducing the interpolation artifacts and the MSE values. The isotropic approach described in [83] (only one iteration) gives a bit smaller MSE values than the proposed method, but, on the other hand, a visual comparison of the interpolated images shows that the proposed technique avoids zippering near the edges and preserves with more precision the contours of the images (see Section 3.4).

3.4 Computational cost and results

3.4.1 Computational cost

An analysis of the computational complexity of the proposed approach can be done calculating all the operations required by the procedure described in the previous sections. The following

Test-image		Before correction	After correction described in [83]	After proposed correction
lighthouse	R	9.70	8.14	8.21
	G	6.93	3.74	4.33
	B	7.68	5.84	6.03
sail	R	5.22	4.77	4.80
	G	3.17	2.07	2.16
	B	3.66	3.26	3.14
boat	R	9.24	6.77	6.90
	G	8.02	3.63	4.32
	B	12.15	9.94	9.89
statue	R	6.23	5.13	5.62
	G	5.17	2.77	3.33
	B	7.77	6.77	7.08
window	R	4.81	4.99	5.03
	G	3.29	2.38	2.50
	B	6.13	6.76	6.56
Average	R	7.04	5.96	6.11
	G	5.32	2.92	3.33
	B	7.48	6.51	6.54

Table 3.4: *MSE comparison of different approaches for refining the high-frequency values.*

operations are considered: additions, multiplications, absolute value computations, bit-shifts and comparisons. The multiplications with powers of two are substituted with bit-shifts.

Denoting with M and N the width and the height of the image, each directional interpolation of the green channel requires 4 additions and 3 bit-shifts for an half of the MN pixels of the image, while the decision step can be implemented with 18 additions, 2 absolute value computations and 1 comparison for each one of the $MN/2$ estimated values. So, the reconstruction of the green channels requires $13MN$ additions, MN absolute value computations, $3MN$ bit-shifts and $0.5MN$ comparisons. Both the red and the blue interpolation need $3MN$ additions, $0.75MN$ bit-shifts and $0.25MN$ comparisons. Therefore producing a full color interpolated image requires 19 additions, 1 absolute value, 4.5 bit-shifts and 1 comparison per pixel.

The refining step described in Section 3.3 needs 7.5 additions, 1.5 bit-shifts, 0.5 multiplication and 1 comparison extra-operations per pixel. This step can be performed in real time when the computational resources of the camera allow the corresponding increase in complexity or as a post-processing if the Bayer data are stored.

In Table 3.5 the computational cost of the proposed algorithm is compared with those of the demosaicking techniques described in [52, 55, 64]. The proposed scheme is very less demanding than other high-quality approaches (even including the additional refining step) and, as we shall see in the next section, this is not paid with a reduction in performance.

3.4.2 Experimental results

Table 3.6 reports the experimental results of the proposed algorithm and other demosaicking approaches. The *color peak signal-to-noise ratio* (CPSNR) is considered (see Section 2.10)

	ADDSs	ABSs	SHIFTs	MULTSs	COMPs	LUTs	OPs
Bilinear Interpolation	4		3				7
Edge-Directed Interpolation [55]	16	4	4.5		1		25.5
Demosaicking with POCS [52]	400	4	4.5	384	1		793.5
Homogeneity-Directed Demosaicking [64]	106	12		50	103	6	277
Proposed Algorithm	19	1	4.5		1		25.5
Proposed Algorithm (with refining)	26.5	1	6	0.5	2		36

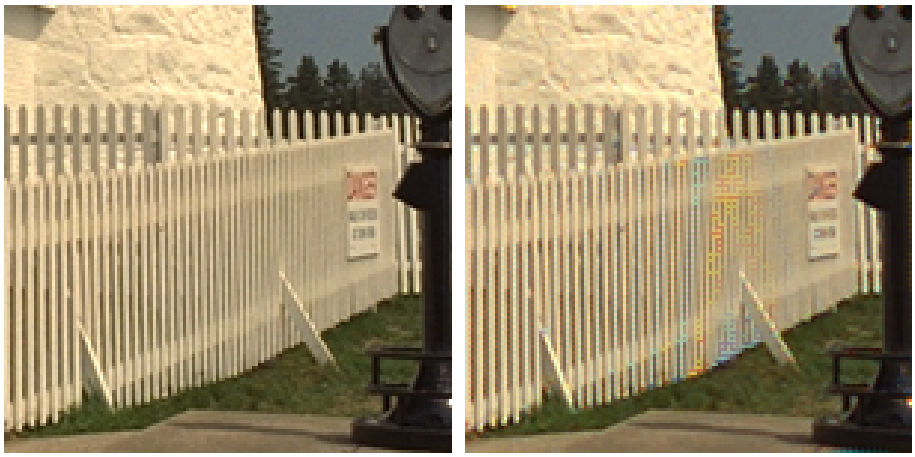
Table 3.5: *Computational cost per pixel of different demosaicking techniques.*

Method	Bilinear	[52]	[55]	[64]	[83]	Proposed
1	26.23	37.90	31.85	35.15	**38.44**	36.91
2	33.09	38.54	37.53	38.88	35.51	**40.16**
3	26.72	37.23	32.83	35.48	35.08	**37.44**
4	27.73	38.68	33.29	37.59	39.14	**39.24**
5	33.46	41.21	39.01	40.33	39.11	**41.61**
6	23.64	35.56	29.88	33.83	**35.66**	35.42
7	32.50	41.71	38.39	40.96	41.46	**42.34**
8	32.30	41.06	37.98	40.56	40.90	**42.14**
9	29.30	**39.34**	34.60	37.59	38.91	39.30
10	32.89	41.86	38.69	41.57	41.65	**42.95**
11	23.95	34.46	28.46	31.45	**35.71**	33.33
12	31.49	38.40	36.10	37.72	37.25	**39.01**
13	31.36	41.84	36.81	41.49	42.14	**43.05**
14	31.96	40.93	36.92	39.31	**41.12**	40.77
15	27.92	**36.50**	32.28	34.12	36.21	36.10
16	28.16	39.80	34.66	38.04	**40.16**	39.82
17	30.29	37.48	33.49	38.55	39.19	**39.87**
18	28.58	38.67	33.68	36.50	**39.13**	37.93
19	30.54	37.73	35.25	36.35	37.17	**37.74**
20	26.64	34.14	30.73	32.79	**34.54**	34.21
Ave.	29.44	38.65	34.62	37.41	38.42	**38.97**

Table 3.6: *CPSNR comparison of different demosaicking methods* (dB).

and the set of 20 Kodak test images reported also in [52] is used. We sampled them according to the Bayer pattern and afterwards we reconstructed them with different demosaicking techniques, comparing the interpolated images with the original ones. The algorithm described in Section 3.2, with the refining step of Section 3.3, is compared to the bilinear interpolation, the edge-directed approach of [55] and the recent demosaicking techniques presented in [52, 64] and [83]. For these three schemes, the MATLAB source code provided us by the authors was used.

The proposed algorithm outperforms the other techniques in the majority of the images and also the CPSNR average is the highest. However, it is known that objective measures, such as MSE and CPSNR, often fail to show the subjective quality of the images and a more effective evaluation is given by the visual inspection of the reconstructed images. In Fig. 3.5 a sample of the image *lighthouse* interpolated with different techniques is shown (this one and other images are available at the web site `http://www.dei.unipd.it/ddfapd`. It can



(a)

(b)

(c)

(d)

(e)

(f)

Figure 3.5: *Portion of the image* lighthouse*: (a) original image; (b) image reconstructed by technique [55]; (c) image reconstructed by technique [64]; (d) image reconstructed by technique [52]; (e) image reconstructed by technique [83]; (f) image reconstructed using the algorithm described in this chapter.*

be noticed that the proposed method presents less aliasing artifacts with respect to the other schemes. Moreover, because of the directional approach, also the zipper effect is avoided near the edges.

To give additional evidence of the performance of the proposed algorithm, we provide also some results obtained by demosaicking raw Bayer data captured using a Nikon D100 camera. This camera, which has a 6.1 megapixels CCD sensor with a Bayer color filter array, allows to store on its memory card the raw data in an uncompressed raw format.

Fig. 3.6 shows a detail of one image demosaicked using the various techniques considered in this chapter. The proposed algorithm and the most recent methods present little aliasing effects also in the high-frequency regions, but some algorithms introduce noticeable zipper effect along the edges of the image. It can be noticed that the proposed technique provides the best compromise between reconstruction of details and absence of zipper effect, also when compared to the most computational demanding methods. Again, this and few other images are available at the web site `http://www.dei.unipd.it/ddfapd`.

3.5 A novel technique for reducing demosaicking artifacts

3.5.1 Introduction

As described in Chapter 2 and in the first part of this chapter, many of the recent demosaicking algorithms share the same method to interpolate the green component from the Bayer data. In fact, the green channel is reconstructed according to the scheme of Fig. 2.2(a), and the direction along which the two filters are applied can be selected using an edge-detection estimator as in [55] or, alternatively, the interpolation can be performed both horizontally and vertically and the decision for the best reconstruction is performed afterwards [p4, 64, 140, 143].

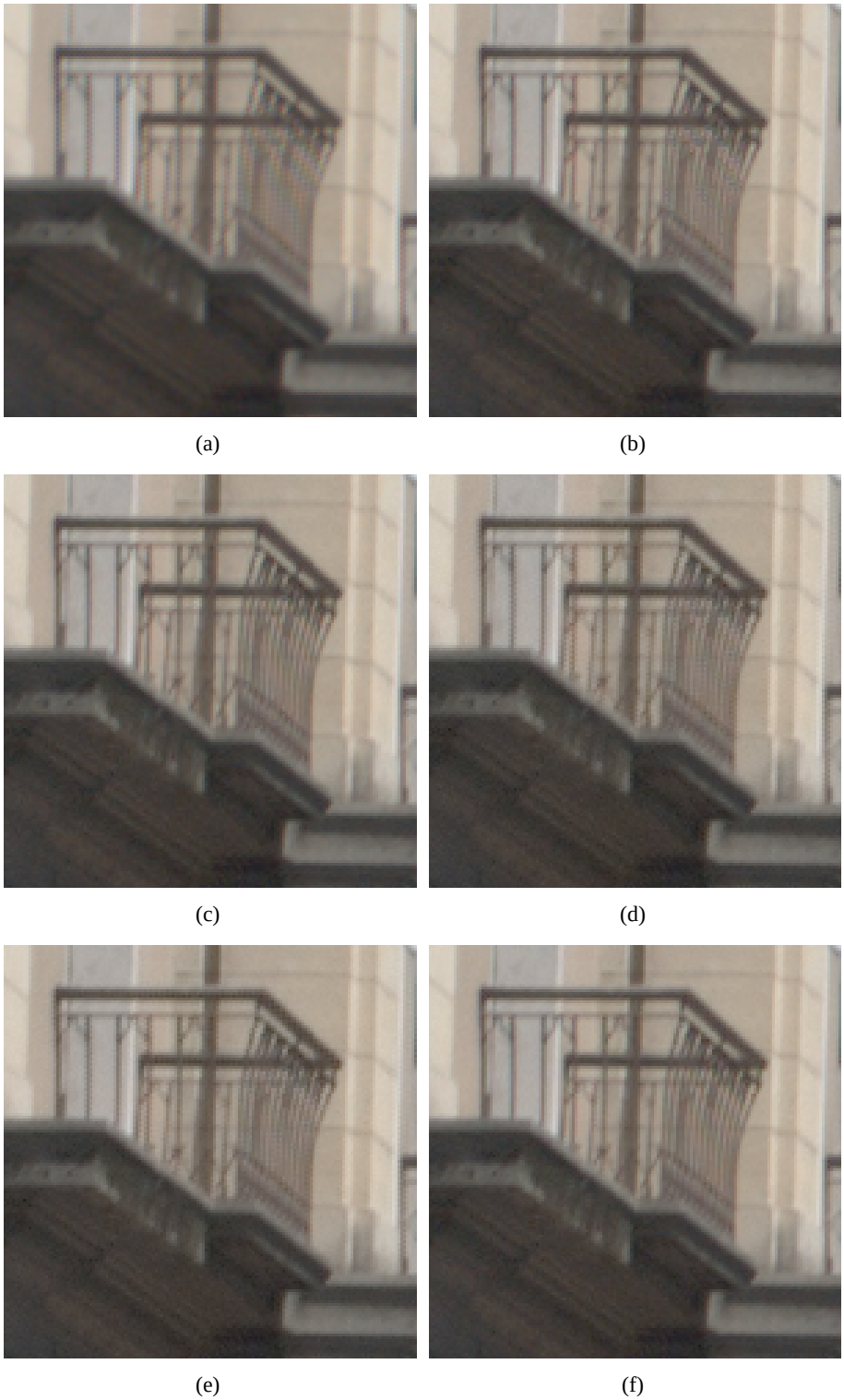
This approach gives good performances and exploits all the information given by the Bayer pattern. However, it cannot avoid some errors in the reconstruction of the full color image, also with the most demanding algorithms.

There are three principal reasons for these reconstruction errors in the interpolation of the green channel:

- a wrong estimation of the direction of the edges. This leads to an incorrect decision for the direction of the filters;
- a local weak correlation in the mid-frequencies between the three color bands. So the term filtered by the filter h_1 and added to the green estimation (see Fig. 2.2(a)) does not estimate correctly the green mid-frequency value;
- the low-pass characteristic of the filter used to interpolate the green values which is not able to provide the high-frequencies of the image.

Since the estimated green image is used to determine the red and blue components also, these misregistrations could be propagated to the other bands of the image.

This leads to introduce a new step to remove the demosaicking artifacts, as much as possible. Gunturk *et al.* [52] and Li [83] propose two iterative algorithms to recover and improve



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

Figure 3.6: *Portion of an image captured with a Nikon D100 digital camera and demosaicked with the following methods: (a) bilinear interpolation; (b) technique [55]; (c) technique [64]; (d) technique [52]; (e) technique [83]; (f) algorithm described in this chapter.*

the interpolated image. However their methods do not use edge information, so they introduce some zipper effect near the contours of the objects, degrading the final quality of the

image. Moreover, because of their iterative nature, they are quite computational demanding, especially [52].

In this section we present an approach for reducing the artifacts introduced by the demosaicking technique following the strategy above described. This approach exploits the high-frequency inter-channel correlation and furthermore avoids to produce zipper effect near the contours of the objects.

3.5.2 Description of the proposed refining algorithm

Edge-direction estimation

As described in the previous section, a key issue of many demosaicking techniques is the edge-directed approach. However, when the edge-estimation is performed in a wrong way, it can introduce some annoying artifacts. Thus, we choose a directional approach for our refining algorithm, but, as a first step, we perform an edge-detection that provides a reliable estimation. We apply the *Sobel operator*, a simple and effective tool for edge-detection in grey-scale images [50, 71]. It is based on two filters, defined as:

$$H_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \qquad H_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}. \tag{3.21}$$

When applied to the image, H_x and H_y give an estimate of the first-order partial derivatives along the horizontal and vertical direction, namely $A_x(n_1, n_2) = H_x * f(n_1, n_2)$ and $A_y(n_1, n_2) = H_y * f(n_1, n_2)$ respectively. The gradient of the image f at the pixel (n_1, n_2) is defined as

$$\nabla_f(n_1, n_2) = \begin{bmatrix} A_x(n_1, n_2) \\ A_y(n_1, n_2) \end{bmatrix} \tag{3.22}$$

and its direction is given by

$$\alpha_f(n_1, n_2) = \arctan\left(\frac{A_y(n_1, n_2)}{A_x(n_1, n_2)}\right) \tag{3.23}$$

where the angle is measured with respect to the x-axis. The direction of an edge is perpendicular to the direction of the gradient vector.

We choose two directions for the application of the directional filtering, precisely the two cardinal directions. Therefore the angle α_f has to be rounded and two main states are considered: $\alpha_f \simeq 0$ (for vertical edges) and $\alpha_f \simeq \pm\pi/2$ (for horizontal edges). If $\alpha_f \simeq 0$ the refining is performed vertically, otherwise horizontally, as represented in Fig. 3.7. Moreover, considering that the arctangent is a monotonically increasing function, the edge-detection can be simplified as follows:

```
if |A_h| > |A_v|
then
 apply vertical filtering
```

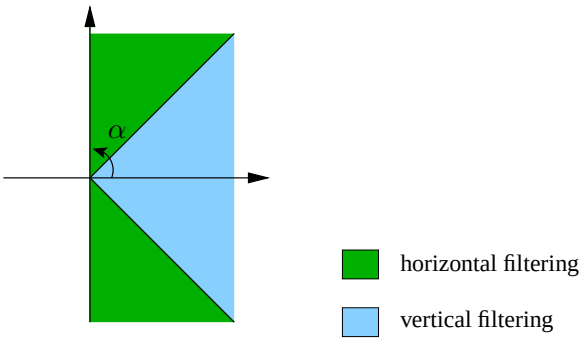


Figure 3.7: *Selection of the directional filtering basing on the values of* α.

```
else
 apply horizontal filtering
end
```

A possible extension of this process of edge-detection could consider four principal directions, instead of only two, i.e. selecting also the two cardinal directions associated to $\alpha_f \simeq \pi/4$ and $\alpha_f \simeq -\pi/4$.

We choose to perform the edge-detection by applying the Sobel operators over the green image because it is the most reliable reconstructed channel. Since the three color channels are very correlated, they present similar edges, so the same edge-estimation can be used for all them.

Refining of the green component

As seen in Section 3.5.1, the filters applied to reconstruct the green band can introduce some errors. Furthermore, using the green channel to reconstruct the red and blue colors, these errors are propagated to the other bands. We can correct them by exploiting the inter-channel correlation of the three primary colors. The three reconstructed full-dimension components are used and, moreover, the knowledge of which pixels have been acquired from the sensor according to the Bayer pattern is exploited.

The first step consists in the correction of the green component, using the edge-direction estimation performed on the image with the Sobel operators. As described in Section 3.5.1, some errors derive from the low-pass characteristic of the filters applied in the reconstruction, which do not consider the high-frequency content. However, the lost high-frequency information can be estimated using the high-bands correlation between the colors. In fact, a solution consists in replacing the high-frequency component of each estimated green value with the one of the color of the Bayer-sampled data at the same pixels. The low-frequency component is preserved unchanged because the low bands are less correlated.

Then, for a pixel at location (n_1, n_2), the green value is decomposed as

$$G(n_1, n_2) = G^{\ell}(n_1, n_2) + G^{h}(n_1, n_2), \tag{3.24}$$

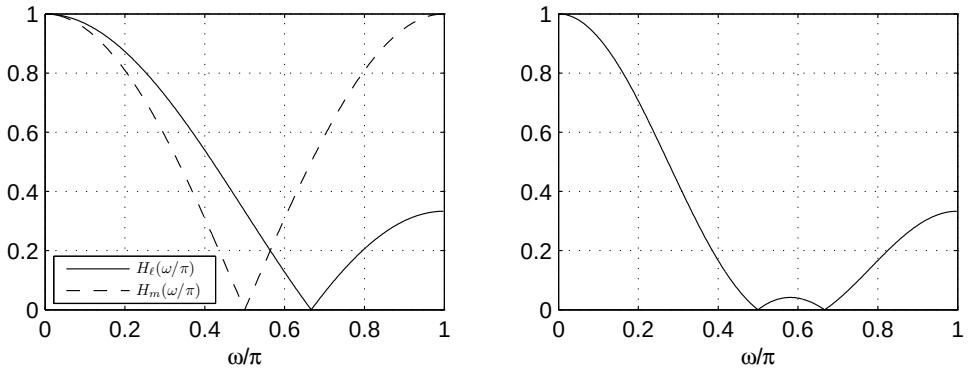


Figure 3.8: *(a) Frequency response of the two filters h_ℓ and h_m; (b) frequency response of the filter h_u.*

where $G^\ell(n_1, n_2)$ and $G^h(n_1, n_2)$ denote the low- and high-frequency components, respectively. Supposing that at this pixel the red value is known from the sensor, also the red value is decomposed into the two components $R^\ell(n_1, n_2)$ and $R^h(n_1, n_2)$. Then, the green high-frequency value $G^h(n_1, n_2)$ is replaced with $R^h(n_1, n_2)$ and the green sample is reconstructed as

$$G(n_1, n_2) = G^\ell(n_1, n_2) + R^h(n_1, n_2). \tag{3.25}$$

The correction at pixels where the blue values are known is carried out in a similar way. The selection of the low-frequencies is performed using a low-pass filter, while the high-frequencies are calculated subtracting the low-frequency values. The low-pass filter is designed as a 1-dimensional filter, in order to avoid the introduction of artifacts near the edges, and the filter direction is driven by the value of the gradient angles $\alpha_f(n_1, n_2)$. The applied filter is the 3-tap FIR $h_\ell = [1/3, \ \ 1/3, \ \ 1/3]$ (see Fig. 3.8(a)).

Another misregistration in the interpolated images can be found in the mid frequencies due to an incidental wrong effect of the filter h_1 in the scheme of Fig. 3.3; it is a consequence of the weak correlation between the three bands when there is a wrong selection of the filtering direction. It can be corrected by replacing the mid-frequencies of the green channel with the ones of the red or blue channels along the direction selected with the edge-estimation, as for the correction of the high-frequencies. A good filter to suppress the mid-frequency terms is the FIR $h_m = [1/2, \ \ 0, \ \ 1/2]$ (see Fig. 3.8(a)).

We note that the cascade of the two filters h_ℓ and h_m corresponds to a single filter h_u with coefficients $[0.17, \ \ 0.17, \ \ 0.32, \ \ 0.17, \ \ 0.17]$ having the frequency response of Fig. 3.8(b). Therefore we choose to apply directly the filter h_u on the green image.

Refining of the red and blue components

After the refining of the green channel, we correct the other color bands. In this case it is experimentally proved that it is more useful to proceed with a different approach with respect to the correction of the green component.

The green pixels (i.e. the pixels where the sensor captured the green values) present only

Test-image	Bil. Int.	Method [p4]	Method [55]	Method [64]	Method [140]
lighthouse	28.16	39.04	34.66	37.87	37.66
sail	32.50	42.10	38.39	41.21	40.81
boat	27.73	38.22	33.29	37.37	36.87
statue	31.96	40.27	36.92	39.33	39.11
window	33.46	41.29	39.00	41.02	40.69
Average	30.76	40.18	36.45	39.36	39.03

Table 3.7: *CPSNR (dB) for some demosaicking algorithms: Bilinear Interpolation, Demosaicking with Directional Filtering and a posteriori Decision [p4], Edge-Directed Interpolation [55], Homogeneity-Directed demosaicking [64], and Primary-Consistent Soft-Decision demosaicking [140].*

two neighboring red and blue pixels, placed either horizontally, either vertically. Therefore no edge-estimation is used and the correction of the red (blue) values in the green positions is performed using the bilinear interpolation of the color differences $R-G$ ($B-G$) of the pixels where the red (blue) color is known from the Bayer pattern.

Instead, in the blue (red) position we have four red (blue) neighboring pixels, placed in the diagonal directions. Interpolating along the diagonal directions can be counterproductive, for two reasons: there is a more relevant presence of cardinal edges in natural images with respect to the other directions, and it is more demanding to implement another edge-direction estimator that detects the diagonal orientations. So, we prefer to perform the correction along the cardinal directions, selected with the edge-detector above described. Once the direction is chosen, a bilinear interpolation of the color difference $R-B$ is carried out; the color difference $R-B$, instead of $R-G$, is preferred in order to exploit the information of the known color value at each pixel, as explained in Section 3.2.3. Moreover, the correction of the red (blue) component in the blue (red) pixels follows the refining in the green position, so the values already corrected can be used in order to determine a more reliable color difference.

3.5.3 Results

We report in Table 3.7 the *color peak signal-to-noise ratio* (CPSNR) obtained by the application of some demosaicking methods presented in Chapter 2, namely the bilinear interpolation, the directional strategy proposed in [p4] and described in the first part of this chapter, the edge-directed approach of Hamilton and Adams [55], the homogeneity-directed demosaicking [64] (without the median-based enhancement), and the primary-consistent soft-decision demosaicking proposed in [140]. The CPSNR given by the same methods followed by the described post-processing procedure is reported in in Table 3.8. Comparing the results with the ones of Table 3.7, we can observe that there is a noticeable reduction of the error. In Table 3.9 the gain given by the application of the proposed algorithm is reported; it has been computed as difference of the CPSNR of the interpolated image before and after the correction step. The improvement of the quality of the image is bigger for the simplest demosaicking techniques (7.24 dB and 3.86 dB for the bilinear interpolation and the edge-directed approach of Hamilton and Adams, respectively), but it is consistent also when the refining step is applied to well-performing methods, such as [64] and [140].

(a) (b) (c)

Figure 3.9: *Portion of image* lighthouse*: (a) original image; (b) image reconstructed with technique [55]; (c) image reconstructed with technique [55] and corrected with the proposed algorithm.*

Test-image	Proposed Refining After				
	Bil. Int.	Method [p4]	Method [55]	Method [64]	Method [140]
lighthouse	35.26	40.32	39.39	40.14	40.26
sail	39.46	42.67	41.99	42.56	42.46
boat	35.14	39.73	38.09	40.00	39.77
statue	39.42	41.34	40.66	41.27	41.21
window	40.61	41.34	41.42	41.51	41.36
Average	38.00	41.08	40.31	41.10	41.01

Table 3.8: *CPSNR (dB) between the original images and the resulting ones after the application of the proposed algorithm to the images reconstructed with different demosaicking methods.*

Test-image	Refining Gain After				
	Bil. Int.	Method [p4]	Method [55]	Method [64]	Method [140]
lighthouse	7.10	1.28	4.73	2.27	2.60
sail	6.96	0.57	3.60	1.35	1.65
boat	7.41	1.51	4.80	2.63	2.91
statue	7.46	1.07	3.74	1.94	2.10
window	7.15	0.05	2.42	0.53	0.67
Average	7.24	0.90	3.86	1.74	1.98

Table 3.9: *CPSNR gain (in dB) given by the application of the proposed algorithm over the images reconstructed with different demosaicking techniques.*

However, a more effective evaluation of the performances is given by a visual inspection of the reconstructed image, because the objective measures, such as MSE and PSNR, often fail to show the subjective quality of the images. An example of the performance of the algorithm is reported in Fig. 3.9 and other images can be downloaded at `http://www.dei.unipd.it/ntrda`. It can be noticed that the proposed method reduces the demosaicking artifacts and, moreover, due to the directional approach, does not introduce zipper effect near the edges.

Chapter 4

Adaptive demosaicking driven by wavelet analysis of the luminance

In this chapter we propose a demosaicking algorithm based on directional filtering, using a novel approach to locate the image details. Edge-estimation is performed on the luminance component which is estimated from the sensor data using an LMMSE approach, and the edge directions are computed exploiting the properties of the wavelet transform. Experimental results proved the effectiveness of this approach, which provides very good performances both in PSNR and visual quality. Part of this chapter has been presented in [p5].

4.1 Introduction

Many demosaicking approaches are based on edge-directed reconstruction, as described in Section 2. However, many of these adaptive methods do not pay adequate attention to the *multiscale* nature of the edges [98, 99]. In fact, it is a usual approach to avoid interpolation across the edges, without considering that artifacts occur if the filter applied for the interpolation does not recover all the frequency information associated with the edge, while the details having a limited frequency support can be easily reconstructed.

In this chapter, we propose a novel approach to directional demosaicking, which is based on a wavelet analysis of the luminance component to drive an adaptive interpolation of the color components of the image that considers the multiscale nature of the edges.

In Section 4.2 some properties of the wavelet transform are reviewed, from which a technique for the estimation of the edges is derived. Section 4.3 describes a LMMSE approach to estimate the full-resolution luminance component, while in Section 4.4 the proposed demosaicking algorithm is presented. Its performances are discussed in Section 4.5.

4.2 Wavelet-based edge-information extraction

Many reconstruction approaches follow an adaptive strategy to extract information from the given data to fill in the unknown pixels values. Adaptive interpolation allows to exploit the local behavior of the image in order to adjust the reconstruction to the discontinuities and

irregularities peculiar to natural images. In such adaptive approaches, image analysis plays a fundamental role since an erroneous estimation can introduce several artifacts, especially near the edges or boundaries between regions.

Information about image singularities can be conveniently gained in a wavelet framework, exploiting the wavelet property of extracting the multiscale information associated with the edges of an image [28, 98, 99]. First, we review some concepts concerning multiscale edge-detection and the wavelet transform. Successively, an algorithm for the edge-information extraction from the available CFA data will be presented.

4.2.1 Background on multiscale edges and wavelet transform

Let $\theta(n)$ be a *smoothing function* (i.e., a function whose integral is unitary and that converges to zero at infinity). We assume that $\theta(n)$ is differentiable and define

$$\psi(n) = \frac{d\theta(n)}{dx}. \tag{4.1}$$

The integral of $\psi(n)$ is equal to zero, $\int_{-\infty}^{+\infty} \psi(n)dn = 0$, hence $\psi(n)$ can be considered a wavelet. We denote with $\psi_s(n)$ the dilation by a scaling factor s of $\psi(n)$,

$$\psi_s(n) = \frac{1}{s}\psi\left(\frac{n}{s}\right), \tag{4.2}$$

and with $\theta_s(n)$ the dilation of $\theta(n)$. The wavelet transform of a function $f(n)$ at scale s and position n is computed by

$$W_s f(n) = f * \psi_s(n). \tag{4.3}$$

From the linearity of convolution and derivation, we have

$$W_s f(n) = f * s\frac{d\theta_s}{dn}(n) = s\frac{d}{dn}\left(f * \theta_s\right)(n). \tag{4.4}$$

This equation shows that the wavelet transform $W_s f(n)$ is equivalent to the first derivative of the signal smoothed at the scale s. So, the extrema of the absolute value of the wavelet transform correspond to the inflection points of $f * \theta_s$ and, in particular, the maxima of $|W_s f(n)|$ are in correspondence of the sharp variation points of $f * \theta_s$. It is proved in [99] that this result extends to the discrete wavelet transform, such as the dyadic wavelet transform where $s = 2^j$.

The same property holds also for an image, which is a bidimensional signal $f(n_1, n_2)$, where the wavelet transform can be applied by defining two wavelet functions $\psi^H(n_1, n_2)$ and $\psi^V(n_1, n_2)$ such that

$$\psi^H(n_1, n_2) = \frac{\partial\theta(n_1, n_2)}{\partial n_1}, \qquad \psi^V(n_1, n_2) = \frac{\partial\theta(n_1, n_2)}{\partial n_2}. \tag{4.5}$$

Applying $\psi^H(n_1, n_2)$ and $\psi^V(n_1, n_2)$ over $f(n_1, n_2)$ we obtain:

$$\begin{bmatrix} W_s^H f(n_1, n_2) \\ W_s^V f(n_1, n_2) \end{bmatrix} = s \begin{bmatrix} \frac{\partial}{\partial n_1}\left(f * \theta_s\right)(n_1, n_2) \\ \frac{\partial}{\partial n_2}\left(f * \theta_s\right)(n_1, n_2) \end{bmatrix} = s\vec{\nabla}(f * \theta_s(n_1, n_2)), \tag{4.6}$$

where $\vec{\nabla}(\cdot)$ is the gradient vector of the image, smoothed by θ_s, as proposed by Canny in [19]. The modulus of this gradient vector is proportional to the wavelet transform modulus

$$M_s f(n_1, n_2) = \sqrt{|W_s^H f(n_1, n_2)|^2 + |W_s^V f(n_1, n_2)|^2} \tag{4.7}$$

and its angle is given by

$$\beta_s(n_1, n_2) = \tan^{-1}\left(\frac{W_s^V f(n_1, n_2)}{W_s^H f(n_1, n_2)}\right). \tag{4.8}$$

The angle $\beta_s(n_1, n_2)$ indicates the direction along which $f(n_1, n_2)$ has the sharpest variation. Edges are defined as points where the modulus of the gradient vector is maximum in the direction towards with the gradient vector points in the image plane.

The notion of the gradient vector defined in (4.6) is strictly related to the considered scale s. Therefore, it is useful to analyze how the wavelet transform evolves at different scales. For the sake of simplicity, we consider a 1-D signal, since the extension to two dimensions is straightforward.

The extrema points of the wavelet transform propagate across scale and their evolution characterizes the local Lipschitz regularity of the signal. In fact, a function $f(n)$ is uniformly Lipschitz α over an interval (a, b) if and only if there exists a constant $K > 0$ such that, for all $n \in (a, b)$, the wavelet transforms satisfies [99]

$$|W_s f(n)| \leq K s^{\alpha}. \tag{4.9}$$

Therefore, in correspondence of an isolated singularity, we find extrema points at each scale. The amplitude of the extrema increases when the scale s increases if the function $f(n)$ is locally Lipschtz with $\alpha > 1$, remains constant if $\alpha = 0$, or decreases if $\alpha < 0$.

However, a singularity in the point n_0 does not affect only the value of $W_s f(n_0)$ but its influence is extended to a neighborhood of n_0. Working with a wavelet with a compact support equal to $[-C, C]$, the neighboring points n which wavelet transforms are affected by the singularity in n_0 are included in the *cone of influence* [98] defined by

$$|n - n_0| \leq Cs. \tag{4.10}$$

Therefore identifying non-isolated singularities becomes more difficult at high scales with respect to the lowest values of s because the cones of influence are wider and intersect one another. This is evident in Fig. 4.1 that shows a portion of the test image *lighthouse* and the wavelet transform of the green component along the marked row. The discrete dyadic wavelet transform proposed in [99] is applied at the scales $s = 2^j$, for $j = 1, 2, 3$. In the first part of the signal some isolated singularities are encountered and the extrema points of the wavelet transform propagate across scale according to the power law denoted by the Lipschitz regularity. Instead, in the right part of the image the singularities are closely spaced and their presence is visible only at the finest scales because when the scale increases the cones of influence intersect one another and the effects of the extrema propagation become negligible.

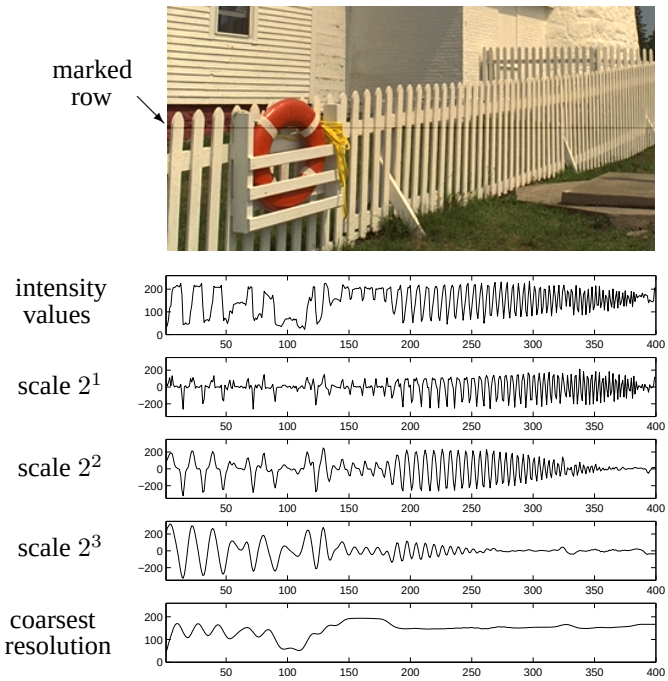


Figure 4.1: *Portion of the test image lighthouse and the dyadic wavelet transform of the intensity of the green component along the marked row. The details for three scales and the coarse signal are shown.*

4.2.2 Edge-detection algorithm

In the previous section we observed that each type of edge presents a different frequency spectrum. In fact the behavior of the edges varies at different scales depending upon the local regularity quantified by the Lipschitz exponent α. Some discontinuities, such as a step-like singularity which has Lipschitz exponent $\alpha = 0$, are visible at each scale and their energies are evenly distributed in all the regions of the spectrum. On the other hand, some singularities present a different amount of energy in the various scales. An example is given in Fig. 4.1, where the details reported in the right part of the figure have all the energy located in the finest scales only, while the energy of the edges placed on left is distributed over large part of the spectrum.

Since the reconstruction of a sampled signal usually is performed using a low-pass filter with a limited frequency support to remove the aliasing component, the edge information contained in the finest scales cannot be recovered. The error that occurs between the original signal and its estimate has energy proportional to the energy of the finest scales which are not included in the pass band of the filter and produce aliasing replicas overlapping the frequency support of the filter. In a bidimensional signal, such as an image, this reconstruction error can be reduced with an edge-directed approach, estimating in which direction the energy of the finest scales is lower and filtering only in this direction. In this way the loss of details is small and a good estimate of the original image can be obtained.

As described in Chapters 2 and 3, the most used filter for the reconstruction of the green component from the available CFA samples is the 5-tap FIR filter proposed by Adams [3]. It

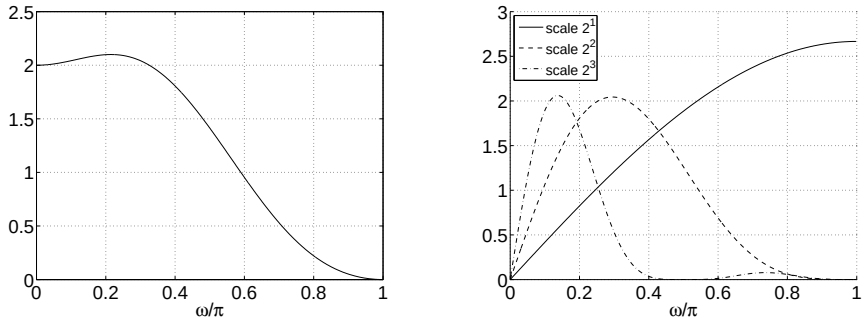


Figure 4.2: *(a) Frequency response of the* 1*-D filter used to estimate the green component; (b) frequency responses of the filters used to obtain the wavelet transforms at the first three scales, with the dyadic wavelet transform proposed in [99].*

is obtained by the combination of a low-pass filter having coefficients $[0.5\ 1\ 0.5]$ applied on the green component and an high-pass filter $[-0.25\ \ 0\ \ 0.5\ \ 0\ \ -0.25]$ used on the red (or blue) pixels to improve the estimation of the missing green values, exploiting the correlation existing between the color components (see Chapter 3 for the details). The frequency response of this filter is shown in Fig. 4.2(a), while in Fig. 4.2(b) the frequency responses of the filters used to obtain the wavelet transforms at different scales are given. It can be observed that the frequencies corresponding to the coarse scales are included in its frequency support, while the filter is unable to detect the discontinuities belonging to the finest scale $s = 2^1$. Therefore, in order to reduce the losses due to the low-pass characteristic of the filter, it is convenient to perform the edge-detection in this scale and apply the filter along the direction d, with $d = H, V$, for which the energy of $W^d_{2^1} f(n_1, n_2)$ is the lowest.

Based on these considerations, an analysis filter bank is applied to signal $f(n_1, n_2)$ and its wavelet transform is computed by

$$\begin{aligned} W^H_{2^1} f(n_1, n_2) &= f * \psi^H_{2^1}(n_1, n_2) \\ W^V_{2^1} f(n_1, n_2) &= f * \psi^V_{2^1}(n_1, n_2), \end{aligned} \tag{4.11}$$

where $\psi^H_{2^1}(n_1, n_2)$ and $\psi^V_{2^1}(n_1, n_2)$ are two wavelet functions defined as in Section 4.2.1. Different classes of wavelet bases can be used. In this chapter we consider the quadratic splines wavelet proposed in [99], however the performance of the algorithm does not show meaningful differences if other bases are used.

Then, the energies of the horizontal and vertical details are computed,

$$\begin{aligned} e_h(n_1, n_2) &= |W^H_{2^1} f(n_1, n_2)|^2 \\ e_v(n_1, n_2) &= |W^V_{2^1} f(n_1, n_2)|^2. \end{aligned} \tag{4.12}$$

The values of $e_h(n_1, n_2)$ and $e_v(n_1, n_2)$ give a measure of the amount of details of the original image that the 1-D low-pass filter is unable to reconstruct when it is applied horizontally or vertically, respectively. If $e_h(n_1, n_2) \gg e_v(n_1, n_2)$ it is preferable to estimate the original value in the pixel (n_1, n_2) with a vertical interpolation, if $e_h(n_1, n_2) \ll e_v(n_1, n_2)$ the best reconstruc-

tion is performed with a horizontal filter, while if $e_h(n_1, n_2) \simeq e_v(n_1, n_2)$ the two directional filters have similar performance. Therefore, the preferable direction for the reconstruction in the pixel (n_1, n_2) can be expressed by the angle given by

$$\eta(n_1, n_2)) = \tan^{-1}\left(\frac{\tilde{e}_h(n_1, n_2)}{\tilde{e}_v(n_1, n_2)}\right), \tag{4.13}$$

where $\tilde{e}_h(n_1, n_2) = b_h * e_h(n_1, n_2)$ and $\tilde{e}_v(n_1, n_2) = b_v * e_v(n_1, n_2)$, with b_h and b_v two averaging kernels of dimensions 1×5 and 5×1, respectively. The averaging is used to avoid sharp fluctuations in the value of $\eta(n_1, n_2)$.

This technique can be applied only on a full-resolution image, otherwise the scale $s = 2^1$ cannot be computed. This is not the case of the green component available for the CFA data because it is sampled on a quincunx lattice. An approach to overcome this problem could consists in performing a rough estimation of the missing values, but it cannot produce a reliable estimate of the edges. A more effective solution is to estimate the full resolution luminance component from the CFA-sampled values, as it will be shown in the next section. Since the luminance and the color components are highly correlated [87], the wavelet analysis can be performed on the estimated luminance, and this information used to adaptively reconstruct the full resolution color image. A detailed implementation of this approach is reported in Section 4.4.

4.3 Luminance estimation

Luminance is an important image characteristic for the human visual system and represents the achromatic visual information [6]. CFA-based sensors do not acquire the luminance information, however it can be estimated from the subsampled color values.

Let denote with $\boldsymbol{r}$, $\boldsymbol{g}$ and $\boldsymbol{b}$ the stacking vectors of the three color components of an image, respectively. Then, the luminance vector is defined as [6]

$$\boldsymbol{\ell} = \frac{1}{4}\boldsymbol{r} + \frac{1}{2}\boldsymbol{g} + \frac{1}{4}\boldsymbol{b}, \tag{4.14}$$

and the vector denoting the CFA-sampled data can be expressed as

$$\boldsymbol{i}_s = \boldsymbol{P_r}\,\boldsymbol{r} + \boldsymbol{P_g}\,\boldsymbol{g} + \boldsymbol{P_b}\,\boldsymbol{b} + \boldsymbol{\eta} \tag{4.15}$$

where $\boldsymbol{P_r}$, $\boldsymbol{P_g}$, $\boldsymbol{P_b}$ are diagonal matrices corresponding to the CFA-sampling of the three color components [135], and $\boldsymbol{\eta}$ is an additive white noise that is assumed to be uncorrelated with respect to the color values.

It is found that the best linear estimate of the luminance from the available CFA-data, $\hat{\boldsymbol{\ell}} = \boldsymbol{\mathcal{G}}\,\boldsymbol{i}_s$, that minimizes the average mean square error

$$\sigma_e^2 = \frac{1}{N_1 N_2} E\left[\left(\boldsymbol{\ell} - \hat{\boldsymbol{\ell}}\right)^T \left(\boldsymbol{\ell} - \hat{\boldsymbol{\ell}}\right)\right] \tag{4.16}$$

(where $N_1 N_2$ is the size of $\boldsymbol{\ell}$ and $E[\cdot]$ denotes the expectation operator), is obtained by the

orthogonality principle [71]

$$E\left[\left(\boldsymbol{\ell}-\hat{\boldsymbol{\ell}}\right)\boldsymbol{i}_s^T\right]=0. \tag{4.17}$$

This gives the Wiener filter as the $N_1N_2 \times N_1N_2$ matrix

$$\boldsymbol{\mathcal{G}}=E\left[\boldsymbol{\ell}\,\boldsymbol{i}_s^T\right]\left(E\left[\boldsymbol{i}_s\boldsymbol{i}_s^T\right]\right)^{-1}. \tag{4.18}$$

By introducing the matrices $\boldsymbol{H}=[\boldsymbol{P_r}\ \boldsymbol{P_g}\ \boldsymbol{P_b}]$ and $\boldsymbol{A}=\left[\frac{1}{4}\boldsymbol{I}\ \ \frac{1}{2}\boldsymbol{I}\ \ \frac{1}{4}\boldsymbol{I}\right]$, with $\boldsymbol{I}$ the $N_1N_2 \times N_1N_2$ identity matrix, and denoting with $\boldsymbol{i}$ the vector obtained by stacking the three color component vectors, that is $\boldsymbol{i}^T=[\boldsymbol{r}^T\ \ \boldsymbol{g}^T\ \ \boldsymbol{b}^T]$, relations (4.14) and (4.15) can be expressed as

$$\boldsymbol{\ell} = \boldsymbol{A}\boldsymbol{i} \tag{4.19}$$

$$\boldsymbol{i}_s = \boldsymbol{H}\boldsymbol{i}+\boldsymbol{\eta} \tag{4.20}$$

and therefore

$$E\left[\boldsymbol{\ell}\,\boldsymbol{i}_s^T\right] = E\left[(\boldsymbol{A}\boldsymbol{i})(\boldsymbol{H}\boldsymbol{i}+\boldsymbol{\eta})^T\right]=\boldsymbol{A}\boldsymbol{R_i}\boldsymbol{H}^T \tag{4.21}$$

$$E\left[\boldsymbol{i}_s\boldsymbol{i}_s^T\right] = E\left[(\boldsymbol{H}\boldsymbol{i}+\boldsymbol{\eta})(\boldsymbol{H}\boldsymbol{i}+\boldsymbol{\eta})^T\right]=\boldsymbol{H}\boldsymbol{R_i}\boldsymbol{H}^T+\boldsymbol{R_\eta} \tag{4.22}$$

where $\boldsymbol{R_i}$ and $\boldsymbol{R_\eta}$ are the autocorrelation matrices of the original image $\boldsymbol{i}$ and of the noise $\boldsymbol{\eta}$, respectively. Therefore the best linear estimate of the luminance is given by the Wiener filter expressed by the matrix

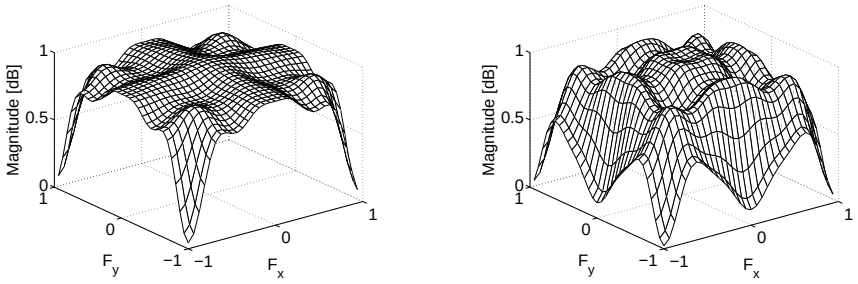
$$\begin{aligned}\boldsymbol{\mathcal{G}} &= \boldsymbol{A}\boldsymbol{R_i}\boldsymbol{H}^T\left(\boldsymbol{H}\boldsymbol{R_i}\boldsymbol{H}^T+\boldsymbol{R_\eta}\right)^{-1}\\ &= \boldsymbol{A}\left(\boldsymbol{H}^T\boldsymbol{R_\eta}^{-1}\boldsymbol{H}+\boldsymbol{R_i}^{-1}\right)^{-1}\boldsymbol{H}^T\boldsymbol{R_\eta}^{-1}.\end{aligned} \tag{4.23}$$

Assuming a gaussian noise with variance $\sigma_{\boldsymbol{\eta}}^2$, we have that $\boldsymbol{R_\eta}=\sigma_{\boldsymbol{\eta}}^2\boldsymbol{I}$. As for the statistical model of a color image, in this chapter we make two assumptions. The first one regards the spatio-spectral correlation between the samples of two different color components X_1 and X_2. A severe but commonly used approximation is to consider the cross-correlation $E[X_1(n_1,n_2)X_2(n_1+m_1,n_2+m_2)]$ as separable into a spectral term and a spatial term [88, 135], that is

$$E[X_1(n_1,n_2)X_2(n_1+m_1,n_2+m_2)]=k_{X_1X_2}R_X(m_1,m_2), \tag{4.24}$$

where $R_X(m_1,m_2)$ is the autocorrelation function of a single color channel, and $k_{X_1X_2}=E[X_1(n_1,n_2)X_2(n_1,n_2)]$ represents the correlation between the components X_1 and X_2. Altough very restrictive, the separability assumption is widely used by many image restoration methods. Therefore the cross-correlation matrix between two color vectors $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ is given by $R_{\boldsymbol{x}_1\boldsymbol{x}_2}=k_{x_1x_2}\boldsymbol{R_x}$, and $\boldsymbol{R_i}$ can be approximated as

$$\boldsymbol{R_i}=\begin{bmatrix}\boldsymbol{R_r} & \boldsymbol{R_{rg}} & \boldsymbol{R_{rb}}\\ \boldsymbol{R_{rg}} & \boldsymbol{R_g} & \boldsymbol{R_{gb}}\\ \boldsymbol{R_{rb}} & \boldsymbol{R_{gb}} & \boldsymbol{R_b}\end{bmatrix}$$



(a) (b)

Figure 4.3: *Frequency response of the filters for the luminance estimation in the green pixels, and in the red/blue pixels, respectively, with size* $M_h = M_v = 9$.

$$= \begin{bmatrix} 1 & k_{rg} & k_{rb} \\ k_{rg} & 1 & k_{gb} \\ k_{rb} & k_{gb} & 1 \end{bmatrix} \otimes \boldsymbol{R_c}, \tag{4.25}$$

where $\otimes$ denotes the Kronecker product. The second assumption regards the correlation of each color channel. In literature, images are often modeled using a 2-D stationary separable Markov process [71], that is

$$R_X(m_1, m_2) = \sigma^2 \rho_h^{|m_1|} \rho_v^{|m_2|}, \tag{4.26}$$

where σ^2 is the variance of the signal, while ρ_h and ρ_v are the row and column correlation coefficients, respectively. In order to give a more accurate representation of the power spectral density of the image, this model can be modified by including also an additive white component [18], giving

$$R_X(m_1, m_2) = \left(\sigma^2 - \sigma_w^2\right) \rho_h H^{|m_1|} \rho_v^{|m_2|} + \sigma_w^2 \delta(m_1)\delta(m_2), \tag{4.27}$$

where σ_w^2 is the variance of the white component.

Due to the data sampling structure of the CFA described in Section 2.4, the optimal filter results periodically space-varying and two different states are observed, giving the two frequency responses reported in Fig. 4.3. As reported in Section 2.4, in the quincunx-placed green pixels the color difference terms modulated at $(0, \pm\pi)$ and $(\pm\pi, 0)$ (see Fig. 2.4(a)) vanish, hence only the chrominance components modulated at $(\pm\pi, \pm\pi)$ have to be eliminated, and the frequency response of the filter results as in Fig. 4.3(a). Instead, in the red and blue locations all the chrominance components have to be considered, and the low-pass filter has the frequency response as in Fig. 4.3(b).

In the filter design, the statistical model of (4.27) has been used (with $\rho_h = \rho_v = 0.9$ and $\sigma_w^2 = 0.1\,\sigma^2$), and the spectral correlation parameters were set to $k_{RG} = k_{GB} = 0.9$ and $k_{RB} = 0.8$. These parameter values were chosen on the basis of the statistical characteristics of the test images used for the experiments in Section 4.5. The noise variance was set to $\sigma_{\boldsymbol{\eta}}^2 = 0.01$.

4.4 Proposed algorithm

The algorithm starts with the estimation of the luminance from the CFA image, using two 9×9 FIR filters designed with the approach described in Section 4.3, whose frequency responses are shown in Fig. 4.3. This full-resolution component is used for the edge-detection approach proposed in Section 4.2.2, obtaining the angle $\eta(n_1, n_2)$ as in (4.13). This gives an estimation of the direction of the edges near the location (n_1, n_2) and is used to compute two classifiers of the edge-direction at each pixel:

$$\begin{aligned} w_h(n_1, n_2) &= 1 - \frac{2}{\pi}|\eta(n_1, n_2)| \\ w_v(n_1, n_2) &= \frac{2}{\pi}|\eta(n_1, n_2)| \ . \end{aligned} \tag{4.28}$$

The two classifiers w_h and w_v are used to adaptively interpolate the green component by estimating the missing green samples as

$$\hat{G}(n_1, n_2) = w_h(n_1, n_2)\hat{G}_h(n_1, n_2) + w_v(n_1, n_2)\hat{G}_v(n_1, n_2) \tag{4.29}$$

where $\hat{G}_h(n_1, n_2)$ and $\hat{G}_v(n_1, n_2)$ are the horizontal and vertical interpolations of the green channel obtained with the 5-taps filter proposed by Adams [3] (see Section 4.2.2). In presence of horizontal details, that is when $w_v(n_1, n_2) < \varepsilon$, with ε small, w_v is rounded to zero and w_h to one, in order to discourage interpolation across edges. The same procedure is performed for the vertical details. In general, a value for the constant ε that avoids interpolation across edges is $\varepsilon = 0.2$. In this way, near an horizontal or vertical edge the interpolation is performed along the same direction of the edge, otherwise both the cardinal directions are considered.

After the estimation of the green channel, the red and blue components are reconstructed through interpolation of the color differences. As a first step, the red and blue channels are filled in the green locations through bilinear interpolation of $R - \hat{G}$ and $B - \hat{G}$ respectively, and then the red samples in the blue locations are estimated as

$$\begin{aligned} \hat{R}(n_1, n_2) &= B(n_1, n_2) \\ &\quad + \frac{w_h(n_1, n_2)}{2}\left(\hat{D}^{RB}(n_1 - 1, n_2) + \hat{D}^{RB}(n_1 + 1, n_2)\right) \\ &\quad + \frac{w_v(n_1, n_2)}{2}\left(\hat{D}^{RB}(n_1, n_2 - 1) + \hat{D}^{RB}(n_1, n_2 + 1)\right) \end{aligned} \tag{4.30}$$

where $\hat{D}^{RB}(n_1, n_2) = \hat{R}(n_1, n_2) - \hat{B}(n_1, n_2)$. The same strategy is applied for the reconstruction of the blue component in the red locations.

Once the full-color image is reconstructed, the final quality can be improved suppressing visible artifacts residing in the demosaicked image. A technique that proved to be effective is based on a median filter applied to the color differences $\hat{R}-\hat{G}$ and $\hat{B}-\hat{G}$, because removes such demosaicking artifacts as false colors and zipper effect, while preserves the image features [64, 89]. Then, we propose to apply three iterations of the median filter, keeping unchanged the values obtained by the CFA, as described in [89].

Method	[p4]	[52]	[87]	[64]	[143]	Proposed Method
1	36.91	37.90	37.58	35.15	38.36	**39.41**
2	40.16	38.54	40.26	38.90	**40.42**	39.61
3	37.44	37.23	38.12	35.50	38.05	**38.27**
4	39.24	38.68	38.11	37.63	40.03	**40.56**
5	41.61	41.21	**42.69**	40.31	42.03	41.95
6	35.42	35.56	35.37	33.81	35.96	**37.10**
7	42.34	41.71	42.72	40.96	42.76	**43.07**
8	42.14	41.06	**42.69**	40.62	41.77	42.62
9	39.30	39.34	39.47	37.59	40.10	**40.62**
10	42.95	41.86	42.77	41.64	42.81	**43.47**
11	33.33	34.46	33.89	31.46	34.73	**35.99**
12	39.01	38.40	**39.53**	37.78	39.17	39.19
13	43.05	41.84	41.30	41.49	43.26	**44.23**
14	40.77	40.93	41.58	39.39	41.43	**42.10**
15	36.10	36.50	36.76	34.18	36.48	**37.55**
16	39.82	39.80	40.01	38.06	40.46	**40.83**
17	39.87	37.48	40.54	38.64	37.80	**40.90**
18	37.93	38.67	38.70	36.55	38.93	**39.93**
19	37.74	37.73	**38.64**	36.46	38.55	38.28
20	34.21	34.14	34.64	33.10	34.91	**35.08**
Ave.	38.97	38.65	39.27	37.46	39.40	**40.04**

Table 4.1: *CPSNR (*dB*) for different demosaicking methods .*

4.5 Experimental results

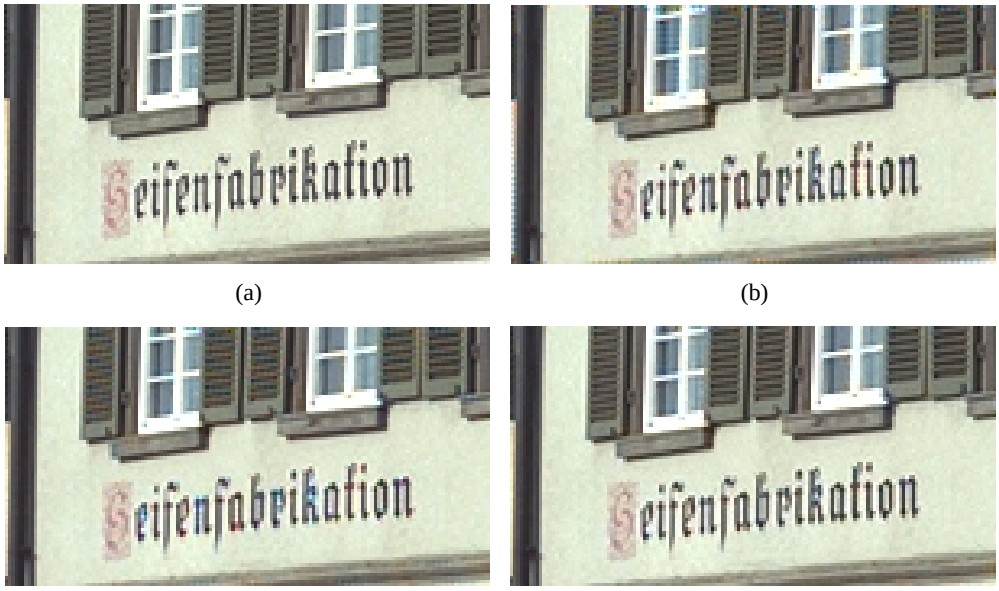
To evaluate the performance of the proposed method, we compare its experimental results with those obtained by other recent demosaicking techniques.

We consider the 20 test images belonging to the Kodak database, used also in Chapter 3, with the methodology described in Section 2.10. Table 4.1 reports the resulting CPSNR for the algorithms presented in [p4, 52, 64, 87, 143] (using the MATLAB code provided by the authors) and for the proposed approach. Moreover, in order to give an evaluation closer to the human visual perception with respect to the CPSNR measure, also the S-CIELAB metric [146] is used (see Table 4.2) to quantify the performance of the different demosaicking algorithms.

We can see that the proposed technique outperforms the other techniques in the majority of the cases with both the metrics. Also the CPSNR and S-CIELAB average values are the highest. We obtain an improvement of 0.77 dB with respect to the method proposed in [87] and of 0.64 dB with respect to the approach of Zhang and Wu [143].

Figures 4.5 and 4.4 show two samples of the images #16 and #6 as interpolated with different methods. It can be noticed that the proposed approach accurately reconstructs even the finest details of the fence and of the writing. Moreover, the false colors introduced due to the aliasing effect are less visible with respect to the images interpolated using other approaches. In fact, the wavelet analysis of the estimated image is able to individuate the high frequencies of the image and drive the adaptive interpolation along the best directions.

Despite its popularity in literature, the film captured images of the Kodak dataset are not very representative of the data acquired by the most recent digital cameras [84]. Therefore we



(a) (b)

(c) (d)

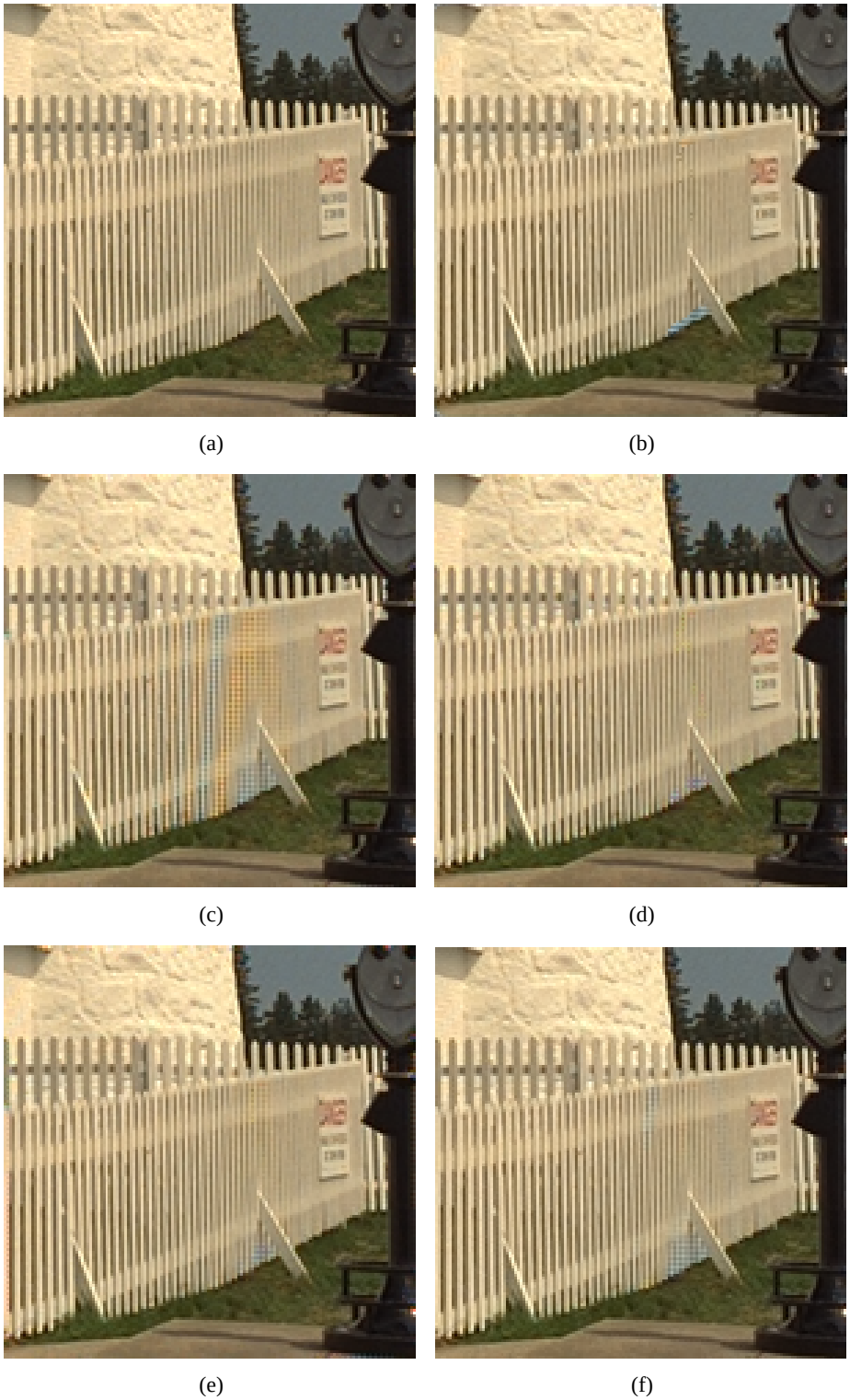
Figure 4.4: *Portion of the image no. 6 of the Kodak set: (a) original image; (b) image reconstructed by technique [143]; (c) image reconstructed by technique [87]; (d) image reconstructed by the proposed approach.*

Method	[p4]	[52]	[87]	[64]	[143]	Proposed Method
1	1.286	1.279	1.205	1.238	1.117	**1.046**
2	0.700	0.806	**0.694**	0.828	0.696	0.786
3	1.118	1.240	**1.004**	1.280	1.079	1.059
4	0.876	1.012	0.955	0.864	0.817	**0.796**
5	0.631	0.689	**0.536**	0.689	0.595	0.646
6	1.450	1.564	1.381	1.412	1.341	**1.248**
7	0.618	0.674	**0.571**	0.632	0.572	0.572
8	0.599	0.637	**0.535**	0.613	0.558	0.552
9	0.792	0.850	0.787	0.837	0.734	**0.717**
10	0.530	0.595	0.509	0.523	0.496	**0.495**
11	1.863	1.805	1.661	1.874	1.553	**1.435**
12	0.756	0.817	**0.674**	0.808	0.724	0.761
13	0.585	0.703	0.684	0.576	0.563	**0.536**
14	0.599	0.633	0.530	0.590	0.558	**0.525**
15	1.165	1.167	**1.058**	1.370	1.084	1.077
16	0.846	0.863	0.782	0.891	**0.766**	0.767
17	0.712	0.732	**0.576**	0.679	0.625	0.660
18	1.028	1.005	0.929	1.026	0.907	**0.869**
19	1.041	1.067	**0.922**	1.191	0.944	1.009
20	1.248	1.250	**1.058**	1.332	1.079	1.129
Ave.	0.922	0.969	0.852	0.963	0.840	**0.834**

Table 4.2: *S-CIELAB error for different demosaicking methods.*

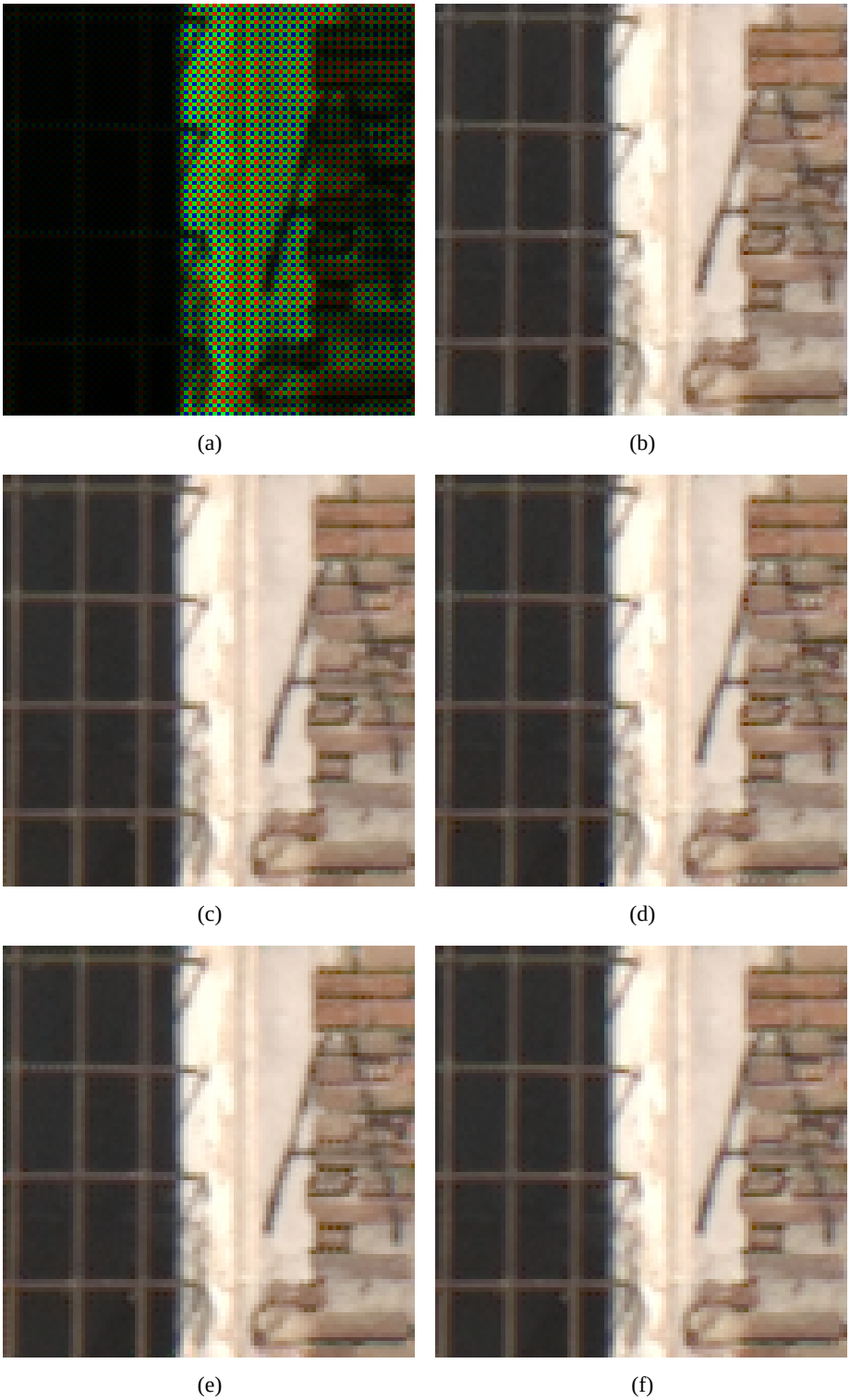
provide also some results obtained by demosaicking raw data captured using a Pentax *ist DS2, with a 6.1 megapixels CCD sensor. Fig. 4.6 shows a portion of one CFA image acquired with

this camera, and the same image interpolated using the techniques presented in [87, 143] and in this chapter. It can be observed that the proposed approach provides the best reconstruction also in the most detailed regions of the image. In particular, no zipper effect or aliasing is introduced near the window or along the lines of the wall.



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

Figure 4.5: *Portion of the image no. 16 of the Kodak set: (a) original image; (b) image reconstructed by technique [64]; (c) image reconstructed by technique [87]; (d) image reconstructed by technique [p4]; (e) image reconstructed by technique [143]; (f) image reconstructed by the proposed approach.*



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

Figure 4.6: *Portion of one image captured with a Pentax *ist DS2: (a) CFA image; (b) image reconstructed by technique [64]; (c) image reconstructed by technique [87]; (d) image reconstructed by technique [p4]; (e) image reconstructed by technique [143]; (f) image reconstructed by the proposed approach.*

Chapter 5

Regularization approaches to demosaicking

In this chapter a novel demosaicking approach is proposed. Since the reconstruction of an image acquired with a color filter array from a mathematical viewpoint is an ill-posed problem, the regularization theory is used. In fact, the role of the regularization is to convert an ill-posed problem into a well-posed one, exploiting the a priori knowledge about the original input signal.

In the first part of this chapter we present a brief survey about regularization. We start considering a problem defined into a couple of functional spaces. Then, we describe several regularization approaches dealing with discrete data that are suitable for different practical applications. Finally, we describe a novel demosaicking technique based on regularization. A preliminary version of this technique has been presented in [p6].

5.1 Overview on regularization

5.1.1 Generalized inverses for problems in functional spaces

Most linear inverse problems assume the following statement [14]: given two functional spaces X and Y, and a linear, continuous operator H from X into Y, the problem is to find, for some $y \in Y$, a function $x \in X$ such that

$$y = Hx. \tag{5.1}$$

This problem is well-posed, in the Hadamard sense, if and only if the operator H is injective, i.e. the equation $Hx = 0$ has only the trivial solution $x = 0$ (*uniqueness*), and it is onto Y (*existence*). Then general theorems of functional analysis ensure that the inverse mapping H^{-1} is also continuous.

On the other hand, if $Hx = 0$ has nontrivial solutions the problem is ill-posed. The set of the solutions of $Hx = 0$ is a closed subspace of X, called the *null space* $N(H)$ of H. Moreover, we assume that the set of the elements of Y that are images of some $x \in X$, named the *range* $R(H)$ of H, is a closed subset of Y. A way to restoring the existence and uniqueness consists in redefining both the solution space X and the data space Y, finding X' and Y', such that for any $y' \in Y'$ there exists a unique x' such that $y' = Hx'$. However this redefinition usually is difficult (or impossible) in practical problems. A method which produces the same

result is the method of the *generalized inverse*.

Least squares solutions

Consider first the case in which the existence condition is not satisfied but H is injective. Then, the functions x that solve the variational problem

$$\hat{x} = \arg\min_x \|Hx - y\|_Y, \tag{5.2}$$

where $\|\cdot\|_Y$ denotes the norm of Y, are called the *least-squares solutions* of the problem (5.1). These solutions can be obtained by solving the Euler equation

$$H^*Hx = H^*y \tag{5.3}$$

where H^* is the adjoint of H [14]. The Euler equation has solution since $R(H)$ is closed.

Generalized solutions

The solution of the Euler equation is not unique when H is not injective, because $N(H)$ is non-trivial. Therefore, when H is not injective, one looks for the solution of the least-square problem which has minimal norm $\|x\|_X$. This solution (denoted by x^+) is unique and is called the *generalized solution*. The generalized solution is orthogonal to $N(H)$. Exploiting its uniqueness, a linear mapping H^+, named *generalized inverse* is defined by

$$x^+ = H^+y. \tag{5.4}$$

The generalized solution x^+ solves (5.1) and is unique. Then it is a well-posed solution.

C-Generalized solutions

In many inverse problems, the generalized solution is trivial or is not meaningful for practical approaches. An extension of the generalized solution looks for minimizing

$$\|Cx\|_Z, \tag{5.5}$$

where C is a linear operator from X into a functional space Z. If there exists a unique least-square solution that minimizes (5.5), we denote it by x_C^+ and we call it a *C-generalized solution*.

Regularization methods

When the range of H is not closed, the inverse H^{-1} or the generalized inverse H^+ is not defined everywhere on Y and it is not continuous. Therefore, both the requirements of existence and continuity do not hold true.

One of the most investigated approach to ill-posed problems is the *regularization method* of Tikhonov [133]. A regularization algorithm for the problem in (5.1) is given in terms of a

one-parameter family of continuous operator P_λ, with $\lambda > 0$, from Y into X such that, for any noise-free $y \in P(H)$

$$\lim_{\lambda \to 0} P_\lambda y = H^+ y = x^+. \tag{5.6}$$

Therefore P_λ provides an approximation of x^+ when $\lambda \to 0$. In case of noisy data,

$$y = Hx + \eta, \tag{5.7}$$

where η represents the noise, we have to consider also the term $P_\lambda \eta$ that typically is divergent when $\lambda \to 0$. Then, a compromise between approximation and noise propagation has to be found. It can be seen also as a compromise between fidelity to the data (given by the least-square solutions) and fidelity to the information expressed in the space Z. The optimal choice of the *regularization parameter* depends on this problem.

Probably the most studied regularization technique consists of minimizing

$$\|H(x) - y\|_Y^2 + \lambda \|Cx\|_Z^2, \tag{5.8}$$

where $\|\cdot\|_Z^2$ denotes the euclidean norm in Z. The solution to this problem is given by $x_\lambda = P_\lambda y$, with

$$P_\lambda = (H^* H + \lambda C^* C)^{-1} H^*. \tag{5.9}$$

Tikhonov proposed to use the following functional,

$$\|Cx\|_Z^2 = \sum_{r=0}^{\gamma} \int c_r(u) |x^{(r)}(u)|^2 du, \tag{5.10}$$

where $x^{(r)}(u)$, for $r = 0, \ldots, \gamma$ indicate the first γ rth-order derivative of $x(u)$, and the weights $c_r(u)$ are strictly positive. Starting from the work of Tikhonov, many regularization techniques have been proposed. Since in image processing usually a discrete model is considered, in the following we review many of them dealing with discrete data, in order to make easy their practical applications to problems concerning images.

5.1.2 Regularized solutions for problems with discrete data

In many inverse problems occurring in the physical science, the following discrete model is considered,

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{\eta}, \tag{5.11}$$

where $\boldsymbol{x}$ is a N_x-length vector representing the original signal, $\boldsymbol{y}$ is the vector containing N_y measured samples, and $\boldsymbol{\eta}$ is an additive random noise. The $N_y \times N_x$ matrix $\boldsymbol{H}$ accounts for the linear operator applied on $\boldsymbol{x}$. The model in (5.11) is usually called the *observation model* and $\boldsymbol{H}$ is indicated as the *degradation* matrix. In image processing problems, usually $\boldsymbol{x}$ and $\boldsymbol{y}$ denote the stacking versions of the bidimensional signals $x(n_1, n_2)$ and $y(n_1, n_2)$, respectively.

When this problem is well-posed, the original signal can be recovered using a least-square approach,

$$\boldsymbol{x} = (\boldsymbol{H}^T \boldsymbol{H})^{-1} \boldsymbol{H}^T \boldsymbol{y}, \tag{5.12}$$

(where $\boldsymbol{H}^T$ denotes the matrix transpose of $\boldsymbol{H}$), or the generalized inverse matrix of $\boldsymbol{H}$, $\boldsymbol{x} = \boldsymbol{H}^+\boldsymbol{y}$.

Problems arise when (5.11) is ill-conditioned, and these solutions are not meaningful for practical applications. A method for dealing with the instability of the problem is that of regularization. Since obtaining the true solution from imperfect data is impossible, the equation (5.11) is considered only to define the admissible solutions $\hat{\boldsymbol{x}}$ such that $\|\boldsymbol{y} - \boldsymbol{H}\hat{\boldsymbol{x}}\| \leq \|\boldsymbol{\eta}\|$, and some *a priori* information about the original signal $\boldsymbol{x}$ is exploited to obtain the *acceptable solution* [34, 115].

Therefore, in the spirit of (5.8), a regularization solution $\hat{\boldsymbol{x}}$ is defined as the solution of the following problem

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \left\{ \Psi(\boldsymbol{x}, \boldsymbol{y}) + \lambda J(\boldsymbol{x}) \right\}. \tag{5.13}$$

The *data fidelity term* $\Psi(\boldsymbol{x}, \boldsymbol{y})$ quantifies the distance between the estimated data and the original one, while the *prior* term (or *regularization term*) is designed to penalize solutions that would not exhibit the expected properties. The *regularization parameter* λ controls the trade-off between the terms in Equation (5.13).

Regularization is widely used in many aspects of image processing, such as denoising, deblurring, segmentation, image magnification, super-resolution, motion estimation, coding, X-ray tomography, etc.

The most popular choice for the distance measure $\Psi(\boldsymbol{x}, \boldsymbol{y})$ corresponds to the least-squares approach of (5.2), using the euclidean norm, that is

$$\Psi(\boldsymbol{x}, \boldsymbol{y}) = \|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|_2^2, \tag{5.14}$$

or the weighted quadratic norm,

$$\Psi(\boldsymbol{x}, \boldsymbol{y}) = \|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|_{\boldsymbol{R}_{\boldsymbol{\eta}}^{-1}}^2, \tag{5.15}$$

where $\boldsymbol{R}_{\boldsymbol{\eta}}$ denotes the autocorrelation matrix of the noise $\boldsymbol{\eta}$ [34].

The design of the prior term is the most relevant task of a regularization problem. In literature, many interesting choices have been examined. Next, we will provide a brief survey of a number of the most popular ones.

Quadratic regularization (C-generalized inverse)

A well-known and widely used regularization technique has been proposed by Tikhonov and Arsenin in [133], using the prior $J(\boldsymbol{x})$ defined as

$$J(\boldsymbol{x}) = \|\boldsymbol{C}\boldsymbol{x}\|_2^2, \tag{5.16}$$

where the $N_y \times N_x$ matrix $\boldsymbol{C}$ is chosen basing on the *a priori* knowledge about $\boldsymbol{x}$. An important requirement is [106]

$$\ker(\boldsymbol{H}^T\boldsymbol{H}) \cap \ker(\boldsymbol{C}^T\boldsymbol{C}) = \{0\}, \tag{5.17}$$

where $\ker(\cdot)$ denotes the kernel of a matrix, otherwise the prior term is not sufficient to convert the ill-posed problem (5.11) into a well-posed one.

Typically in image processing $J(\boldsymbol{x})$ is a smoothness constraint and $\boldsymbol{C}$ represents a high-pass filter, in order to minimize the amount of high-pass energy in the restored signal, since it is known that the spectra of natural images exhibit a low-pass characteristic [9]. One typical choice for $\boldsymbol{C}$ exploits the 2-D laplacian operator given by

$$C = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}. \tag{5.18}$$

Another common choice for $\boldsymbol{C}$ is the identity matrix. In this way, the corresponding constraint forces to minimize the energy of $\boldsymbol{x}$, as in a generalized inverse solution. It is used especially in denoising problems.

Using the prior term (5.16) and the least-squares data-fidelity term, the minimization of (5.13) leads to an equation of the form

$$\left(\boldsymbol{H}^T\boldsymbol{H} + \lambda \boldsymbol{C}^T\boldsymbol{C}\right)\boldsymbol{x} = \boldsymbol{H}^T\boldsymbol{y} \tag{5.19}$$

and the solution is given by

$$\hat{\boldsymbol{x}} = \left(\boldsymbol{H}^T\boldsymbol{H} + \lambda \boldsymbol{C}^T\boldsymbol{C}\right)^{-1}\boldsymbol{H}^T\boldsymbol{y}. \tag{5.20}$$

This direct approach to find the solution is computationally efficient, but its performances are severely dependent on the value of λ. With larger values of λ the solution tends to be over-regularized, and yet with smaller values of λ the restored signal tends to have more noise effects [9]. Moreover, in (5.20) the inversion of a $N_x \times N_x$ matrix is required.

Therefore, iterative approaches have been extensively investigated in literature [9, 75, 76]. Usually iterative techniques solve (5.19) with the method of successive approximations [75]. This leads to the following *Landweber fixed-point* iteration

$$\hat{\boldsymbol{x}}_{k+1} = \hat{\boldsymbol{x}}_k + \beta[\boldsymbol{H}^T\boldsymbol{y} - (\boldsymbol{H}^T\boldsymbol{H} + \lambda\boldsymbol{C}^T\boldsymbol{C})\hat{\boldsymbol{x}}_k]. \tag{5.21}$$

The starting point $\hat{\boldsymbol{x}}_0$ is set to $\mathbf{0}$ and β is known as the *relaxation parameter* that controls the convergence properties. In the case of shift-invariant operators, this iteration converges if

$$0 < \beta < \frac{2}{|\sigma_{\max}|}, \tag{5.22}$$

where $\sigma_{\max}$ is the largest eigenvalue of $(\boldsymbol{H}^T\boldsymbol{H} + \lambda\boldsymbol{C}^T\boldsymbol{C})$ [127].

The use of a quadratic, ℓ_2-based criteria for the data-fidelity and prior terms leads to a linear problem, and thus results in an inverse filter. However, this method has some limitations because it does not permit an adaptive application to the data. It is particularly true for non-stationary signals such as the images, especially near the edges and the details.

Many adaptive approaches for the regularization problem have been proposed to overcome

this drawback. These methods concern different choices of the data-fidelity and prior terms in (5.13). In the following, several interesting choices for adaptive regularization approaches are examined.

Adaptive iteration using the visibility function

The artifacts introduced by non-quadratic regularization are visible especially near the edges, where the high-frequency content makes difficult the correct reconstruction of the signal. Moreover, the human observers are particularly sensible to the visual quality of the edges. On the other hand, psychophysical experiments confirm that noise gives rise to spurious features or texture especially in the flat regions of the image, while its effects are less evident near sharp transitions in image intensity [75].

Therefore, it has been proposed to apply minimizing functionals in (5.13) which behavior takes into account the local feature of the image. Near an edge it is preferable to preserve the image characteristics with respect to force the noise removal, in a flat region the denoising process is more relevant than the fidelity of the reconstruction to the original data. This can be accomplished using the following terms [75, 76],

$$\begin{aligned} \Psi(\boldsymbol{x}, \boldsymbol{y}) &= \|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|^2_{\boldsymbol{W}_1} \\ J(\boldsymbol{x}) &= \|\boldsymbol{C}\boldsymbol{x}\|^2_{\boldsymbol{W}_2}, \end{aligned} \tag{5.23}$$

where $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$ are two diagonal weighting matrices. The entries of $\boldsymbol{W}_1$ are equal to zero in the flat regions and nonzero near the edges, while a complementar behavior is required for the entries of $\boldsymbol{W}_2$. In [75] the diagonal *noise visibility* matrix is used. The diagonal entries v_i of the visibility matrix $\boldsymbol{V}$ are computed according to

$$v_i = \frac{1}{\theta\sigma^2(i) + 1}, \tag{5.24}$$

where θ is a tuning parameter and $\sigma^2(i)$ denotes the local variance of the image in a neighborhood centred around the pixel corresponding to the index i. The resulting values of v_i are mapped into the $[0, 1]$ range. The matrix $\boldsymbol{W}_2$ is set to be equal to $\boldsymbol{V}$, while $\boldsymbol{W}_1 = \boldsymbol{I} - \boldsymbol{V}$, with $\boldsymbol{I}$ denoting the identity matrix.

Then, the solution is found by using the following iteration

$$\hat{\boldsymbol{x}}_{k+1} = \hat{\boldsymbol{x}}_k + \beta[\boldsymbol{H}^T\boldsymbol{W}_1^T\boldsymbol{W}_1\boldsymbol{y} - (\boldsymbol{H}^T\boldsymbol{W}_1^T\boldsymbol{W}_1\boldsymbol{H} + \lambda\boldsymbol{C}^T\boldsymbol{W}_2^T\boldsymbol{W}_2\boldsymbol{C})\hat{\boldsymbol{x}}_k]. \tag{5.25}$$

Total variation regularization

An adaptive technique that has achieved popularity in recent years is the *total variation* (TV) regularization, where the ℓ_1 norm is used [124, 138]. The prior term is defined as

$$J(x) = \|\nabla x(n_1, n_2)\|_1, \tag{5.26}$$

with $\nabla x(n_1, n_2)$ denoting the gradient of the image $x(n_1, n_2)$. Using the vector notation, it can be expressed as

$$J(\boldsymbol{x}) = \sum_{i=1}^{N_x} \sqrt{|\nabla_i^h \boldsymbol{x}|^2 + |\nabla_i^v \boldsymbol{x}|^2}, \tag{5.27}$$

where $\nabla_i^h \boldsymbol{x}$ and $\nabla_i^v \boldsymbol{x}$ are discrete approximations to the horizontal and vertical first order difference, respectively, at the pixel i. However the derivative of (5.27) diverges in zero. To avoid difficulties, in [138] the modification

$$J(\boldsymbol{x}) = \sum_{i=1}^{N_x} \sqrt{|\nabla_i^h \boldsymbol{x}|^2 + |\nabla_i^v \boldsymbol{x}|^2 + \alpha}, \tag{5.28}$$

has been proposed, with $\alpha > 0$, $\alpha \to 0$. This approach is commonly used in literature and leads to the following equation [15, 74, 138]

$$\left(\boldsymbol{H}^T \boldsymbol{H} + \lambda \boldsymbol{S}^T \boldsymbol{W}_{\boldsymbol{x}} \boldsymbol{S}\right) \boldsymbol{x} = \boldsymbol{H}^T \boldsymbol{y}. \tag{5.29}$$

The matrix $\boldsymbol{S}$ is given by

$$\boldsymbol{S} = \begin{bmatrix} \boldsymbol{S}_h \\ \boldsymbol{S}_v \end{bmatrix}, \tag{5.30}$$

where $\boldsymbol{S}_h$, $\boldsymbol{S}_v$ are $N_x \times N_x$ matrices obtained by vertical concatenation of the matrices ∇_i^h and ∇_i^v, respectively, for $i = 1, \ldots N_x$. $W_{\boldsymbol{x}}$ is a diagonal matrix defined as $W_{\boldsymbol{x}} = \text{diag}(\boldsymbol{w}, \boldsymbol{w})$, where $\boldsymbol{w}$ is a $N_x \times 1$ vector with entries

$$w_i = \frac{1}{2\sqrt{|\nabla_i^h \boldsymbol{x}|^2 + |\nabla_i^h \boldsymbol{x}|^2 + \alpha}}. \tag{5.31}$$

Details about the estimation of (5.29) can be found in [138], where the Euler-Lagrange equation related to the total variation problem is solved in the continuous domain and successively a cell-centered finite difference discretization is applied. Recently an alternative approach formulated in the discrete domain has been traced in [15].

Equation (5.29) is nonlinear, since $\boldsymbol{W}_{\boldsymbol{x}}$ depends on $\boldsymbol{x}$. Its solution $\hat{\boldsymbol{x}}$ can be found using the Landweber fixed-point iterative technique,

$$\hat{\boldsymbol{x}}_{k+1} = \hat{\boldsymbol{x}}_k + \beta[\boldsymbol{H}^T \boldsymbol{y} - (\boldsymbol{H}^T \boldsymbol{H} + \lambda \boldsymbol{S}^T \boldsymbol{W}_{\hat{\boldsymbol{x}}_k} \boldsymbol{S})\hat{\boldsymbol{x}}_k]. \tag{5.32}$$

Total variation has been suggested as the "best" regularization prior for image reconstruction problems [74, 124]. In fact, a total variation solution can contain localized steep gradients, so that also the edges are preserved in reconstruction. It was first proposed for image denoising [124], but it has been applied also to deblurring [138], interpolation and super-resolution [43, 44, 104].

Moreover, the total variation regularizer is convex, thus opening the door to the research of efficient algorithms for computing optimal or nearly optimal solutions. This is another of the reasons of its popularity in image processing [15].

Name of the approach	$\phi(t)$	convexity
Tikhonov [133]	t^2	yes
Total variation [124]	t	yes
Vogel and Oman [138]	$\sqrt{t^2+\alpha}$	yes
Perona and Malik [112]	$1-\exp(-t^2)$	no
Geman and Reynolds [47]	$\frac{\alpha t}{1+\alpha t}$	no

Table 5.1: *Summary of some potential functions used in literature.*

In literature other related approaches are been analyzed. In [86], the authors define the prior term as

$$J(\boldsymbol{x}) = \sum_{i=1}^{N_x} \left(|\nabla_i^h \boldsymbol{x}| + |\nabla_i^v \boldsymbol{x}| \right), \tag{5.33}$$

instead of (5.27). In [43, 44] Farsiu *et al.* proposed the bilateral total variation regularization, where

$$J(\boldsymbol{x}) = \sum_{l=-P}^{P} \sum_{m=-P}^{P} \alpha^{|l|+|m|} \|\boldsymbol{x} - \boldsymbol{S}_h^l \boldsymbol{S}_v^m \boldsymbol{x}\|_1. \tag{5.34}$$

The matrices $\boldsymbol{S}_h^l$ and $\boldsymbol{S}_v^m$ shift $\boldsymbol{x}$ by l and m pixels in horizontal and vertical directions, respectively. Therefore $J(\boldsymbol{x})$ corresponds to a sum of finite difference terms, weighted by the scalar values $\alpha^{|l|+|m|}$. It can be observed that this approach is a generalization of the method proposed in [86]. The bilateral total variation regularization has been applied to super-resolution [44], and problems concerning joint demosaicking and super-resolution [43].

Recently Bioucas-Dias *et al.* demonstrated in [15] that the standard total variation term (5.29) can lead to better results than the term (5.33), and the optimality of the total variation regularization over other variational approaches for image restoration and denoising has been proved in [8].

Other edge-preserving regularizations

The total variation regularization was a great success, providing efficient solutions to many inverse problems in image processing. It has led to the development of many other edge-preserving algorithms that can be expressed under an unifying perspective with a prior term of the form

$$J(\boldsymbol{x}) = \sum_{i=1}^{N_x} \phi(\|\nabla_i \boldsymbol{x}\|_2), \tag{5.35}$$

where $\nabla_i = [(\nabla_i^h)^T, (\nabla_i^v)^T]^T$, and $\phi : \mathbb{R}_+ \to \mathbb{R}$ is called a *potential function* (PF). Many different functions can be found in literature, some of the most popular ones are reported in Table 5.1. A general assumption shared by these approaches is that ϕ is continuous and increasing on $\mathbb{R}_+$, with $\phi(t) = 0$ if and only if $t = 0$. The convexity property is important in regards to the minimization process, as it ensures existence and uniqueness of the minimum, however it is not necessary. Note that if $\phi(t) = t^2$, the regularization corresponds to the Tikhonov method, which is obviously not edge-preserving. Also the total variation regularization can be

expressed using (5.35). In this case we have that $\phi(t) = t$. Moreover, the approximation described in (5.28) corresponds to use $\phi(t) = \sqrt{t^2 + \alpha}$. More details about the edge-preserving regularizations can be found in [105, 132].

Half-quadratic regularization

The minimizers $\hat{\boldsymbol{x}}$ of cost-functions such as $\Upsilon(\boldsymbol{x}, \boldsymbol{y}) = \Psi(\boldsymbol{x}, \boldsymbol{y}) + \lambda J(\boldsymbol{x})$ involving adaptive regularization terms as in (5.35) are nonlinear with respect to $\boldsymbol{x}$ and their computation is costly, especially when $\boldsymbol{H}$ is highly ill-conditioned.

In order to reduce the computational cost, an *half-quadratic* formulation of the regularization problem has been proposed [47, 48]. The basic idea is to consider an *augmented cost function* $\Upsilon^*(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{b})$, where $\boldsymbol{b}$ is an auxiliary array of real number, of the form

$$\Upsilon^*(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{b}) = \Psi(\boldsymbol{x}, \boldsymbol{y}) + \lambda \sum_{i=1}^{N_x} \left(Q(\nabla_i \boldsymbol{x}, b_i) + \psi(b_i) \right). \tag{5.36}$$

The function $Q(\cdot, s)$ is quadratic for any $s \in \mathbb{R}$ and $\psi : \mathbb{R} \to \mathbb{R}$ satisfies

$$\min_{s \in \mathbb{R}} \{Q(t, s) + \psi(s)\} = \phi(t) \quad \forall t \in \mathbb{R}, \tag{5.37}$$

where $\phi(t)$ is defined as in the previous paragraph. The last condition ensures that

$$\Upsilon(\boldsymbol{x}, \boldsymbol{y}) = \min_{\boldsymbol{b} \in \mathbb{R}^{N_x}} \Upsilon^*(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{b}), \tag{5.38}$$

that is the new function $\Upsilon^*(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{b})$ *has the same global minimum in* $\boldsymbol{x}$ *as* $\Upsilon(\boldsymbol{x}, \boldsymbol{y})$ [48]. Moreover, $\Upsilon^*(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{b})$ is quadratic in $\boldsymbol{x}$ for each fixed $\boldsymbol{b}$, hence the name of the method. The minimizer $\hat{\boldsymbol{x}}$ is calculated using alternate minimization [106, 107]. If the $(k-1)$th iterate is $(\boldsymbol{b}^{(k-1)}, \boldsymbol{x}^{(k-1)})$, the next one is defined by

$$\boldsymbol{b}^{(k)} = \arg\min_{\boldsymbol{b}} \Upsilon^* \left(\boldsymbol{x}^{(k-1)}, \boldsymbol{y}, \boldsymbol{b} \right) \tag{5.39}$$

$$\boldsymbol{x}^{(k)} = \arg\min_{\boldsymbol{x}} \Upsilon^* \left(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{b}^{(k)} \right). \tag{5.40}$$

Geman and Reynolds [47] first considered quadratic terms of the *multiplicative form*,

$$Q(t, s) = \frac{1}{2} t^2 s \quad t \in \mathbb{R},\ s \in \mathbb{R}_+. \tag{5.41}$$

Later, Geman and Yang [48] proposed also the *additive form*, where

$$Q(t, s) = (t - s)^2 \quad t \in \mathbb{R},\ s \in \mathbb{R}. \tag{5.42}$$

Numerical results have shown that minimization using half-quadratic regularization can speed up computation compared with the steepest descent method [47, 48]. Therefore these fruitful ideas have been extended by many authors [29, 70]. An analysis of these methods and a comparison between multiplicative and additive form is given in [107].

Denoising by wavelet shrinkage

One of the most relevant problems in image processing is the removal of additive white noise from the acquired signal $\boldsymbol{y}$. Many approaches have been proposed for this problem, as mentioned in Section 2.9 and Chapter 6. Many of them are related to the regularization theory. Denoising is considered as the inverse problem of (5.11), where $\boldsymbol{H} = \boldsymbol{I}$, and $\boldsymbol{\eta}$ is the noise that has to be removed. In particular, the total variation criterion has proved to be effective for this problem [124].

Moreover, after the work of Donoho and Johnstone [35, 36], the idea of the wavelet-based shrinkage is become very popular, due to its effectiveness and its simplicity. A wavelet transform is applied to the noisy data, and a shrinkage of the wavelet coefficients is used to remove the noise. This procedure is based on the idea that large coefficients belong to the details of the image, whereas the noise is distributed across small coefficients. Then, the latter are canceled, while the former are kept unchanged (hard-thresholding), or decreased (soft-thresholding). In particular, the hard-thresholding operator is defined by

$$\theta_\mu(a) = \begin{cases} a & \text{if } |a| > \mu \\ 0 & \text{otherwise,} \end{cases} \tag{5.43}$$

and the soft-thresholding operator by

$$\theta_\mu(a) = \begin{cases} a - \mu & \text{if } a > \mu \\ a + \mu & \text{if } a < \mu \\ 0 & \text{otherwise} \end{cases} \tag{5.44}$$

Finally, an inverse wavelet transformation is performed to obtain the denoised image.

This approach can be cast in the regularization framework [42] considering the following prior term,

$$J(\boldsymbol{x}) = \mathbf{1}^T \rho(\boldsymbol{W}\boldsymbol{x}), \tag{5.45}$$

where $\boldsymbol{W}$ is a $N_x \times N_x$ unitary wavelet transform matrix, $\rho(\cdot)$ an arbitrary scalar function, and $\mathbf{1}$ a $N_x \times 1$ vector of ones. Thus

$$J(\boldsymbol{x}) = \sum_{i=1}^{N_x} \rho(\{\boldsymbol{W}\boldsymbol{x}\}_i). \tag{5.46}$$

Typically $\rho(z)$ is assumed to be symmetric and monotonically nondecreasing for $z > 0$. In particular $\rho(z) = |z|^p$ leads to $J(\boldsymbol{x}) = \|\boldsymbol{W}\boldsymbol{x}\|_p^p$. If $p = 2$ it corresponds to a Tikhonov regularization, while $p = 1$ gives $J(\boldsymbol{x}) = \|\boldsymbol{W}\boldsymbol{x}\|_1$. The cost-function,

$$\Upsilon(\boldsymbol{x}, \boldsymbol{y}) = \|\boldsymbol{x} - \boldsymbol{y}\|_2^2 + \lambda \mathbf{1}^T \rho(\boldsymbol{W}\boldsymbol{x}), \tag{5.47}$$

can be expressed as follows, defining $\boldsymbol{x}_w = \boldsymbol{W}\boldsymbol{x}$ and $\boldsymbol{y}_w = \boldsymbol{W}\boldsymbol{y}$,

$$\Upsilon(\boldsymbol{x}_w, \boldsymbol{y}_w) = \|\boldsymbol{W}^{-1}(\boldsymbol{x}_w - \boldsymbol{y}_w)\|_2^2 + \lambda \mathbf{1}^T \rho(\boldsymbol{x}_w). \tag{5.48}$$

Since $\boldsymbol{W}$ is orthogonal and the unitary invariance property holds for the ℓ^2-norm, we obtain

$$\begin{aligned}\Upsilon(\boldsymbol{x}_w, \boldsymbol{y}_w) &= \|\boldsymbol{x}_w - \boldsymbol{y}_w\|_2^2 + \lambda \mathbf{1}^T \rho(\boldsymbol{x}_w) \\ &= \sum_{i=1}^{N_x} \left[(x_w(i) - y_w(i))^2 + \lambda\rho(x_w(i))\right]. \end{aligned} \tag{5.49}$$

Therefore, the minimization problem can be decoupled for each entry $x_w(i)$, and N_x scalar optimization procedures of the form

$$\hat{z} = \arg\min_z (z-a)^2 + \lambda\rho(z), \tag{5.50}$$

with a assumed known, are far easier than the previous N_x-dimensional optimization. The solution to (5.50) is given by

$$\hat{z} = a - \frac{\lambda}{2}\rho'(\hat{z}). \tag{5.51}$$

This optimization applied for any wavelet coefficient $y_w(j)$ allows to estimate the denoised coefficients $\hat{x}_w$, from which the denoised image $\hat{\boldsymbol{x}}$ results after an inverse wavelet transform. In particular, setting $\rho(z) = |z|^0$ (that is using the ℓ_0-norm in the prior term) from (5.51) we obtain the hard-thresholding rule (5.43), while with $\rho(z) = |z|$ (corresponding to the ℓ_1-norm) we find the soft-thresholding operator (5.44), with $\mu = \frac{\lambda}{2}$.

Therefore it has proved that regularization with the prior term shown in 5.45 is equivalent to the wavelet shrinkage as proposed by Donoho and Johnstone.

Many denoising approaches perform shrinkage applying a redundant transformation instead of using a wavelet representation. In fact, it is shown that an overcomplete transform preserves the image characteristics better than the unitary ones. However, in this case the shrinkage rule cannot be obtained by an optimization procedure starting from a prior term $\rho(\boldsymbol{T}\boldsymbol{x})$, where matrix $\boldsymbol{T}$ defines the transformation. Elad [42] proved that the minimization problem associated to a cost-function involving this prior term can be solved with an iterative *Basis Pursuit DeNoising* (BPDN), where at each step a shrinkage operator is applied.

Regularization with multiple prior terms

Regularization can be performed also using several prior terms. In this case the solution is given by

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \left\{ \Psi(\boldsymbol{x}, \boldsymbol{y}) + \sum_n \lambda_n J_n(\boldsymbol{x}) \right\}. \tag{5.52}$$

where $J_n(\boldsymbol{x})$ are the prior terms and λ_n are the regularization parameters that control the trade-off not only with respect to the data fidelity term but also between the different prior terms. The design of the terms and the techniques for the computation of the solution are similar to the case with only one regularization term. However it has to be considered that to find the optimal parameters λ_n becomes more difficult. An approach has been proposed in [13] where a multidimensional extension of the generalized L-curve framework usually applied to find a single parameter is presented.

5.2 Regularization approaches to demosaicking

5.2.1 Introduction

Several demosaicking techniques described in Chapter 2 are designed with explicit reference to the Bayer pattern and, even if many of them are applicable also with other arrangements, rarely can be used with the sensors that acquires color components different from the traditional red, green and blue components, such as in Figg. 2.1(c) and 2.1(d). An approach to reconstruct images acquired with Kodak pattern of Fig. 2.1(c) is described in [56], while methods for the new CFA schemes proposed in [69] are discussed in [68].

Moreover, usually demosaicking approaches do not consider the characteristics of the point spread functions (PSF) of the sensors. However, as it will be demonstrated in this chapter, the impulse responses of the sensors are relevant for the design of the reconstruction algorithm because they affect the spectral overlaps and the correlations of the color components in the sampled image.

In this chapter, we consider a demosaicking method based on *regularization* that is suitable for any type of color arrangements and accounts for the characteristics of the sensor PSFs. Considering demosaicking under a regularization problem perspective can be of relevant interest when a unified approach is required to reconstruct an image that undergoes multiple distorting processes that affect the image quality. As said in Chapter 2, in literature some regularization approaches to demosaicking have been explored in [77], and the total-variation technique is exploited in [125], while Farsiu *et al.* proposed a regularization method combining demosaicking and super-resolution [43].

The problem formulation is presented in the next section. A quadratic regularized solution is reported in Section 5.2.3, while an adaptive reconstruction is described in Section 5.2.4. The choice of the regularization parameters is discussed in Section 5.2.5 and the performances of the proposed approaches are reported in Section 5.2.6.

5.2.2 Problem Statement

Given an original continuous color image $\mathcal{I}_c(\boldsymbol{u}) = [R_c(\boldsymbol{u}),\ G_c(\boldsymbol{u}),\ B_c(\boldsymbol{u})]$, where $\boldsymbol{u} \in \mathbb{R}^2$ and $R_c(\boldsymbol{u})$, $G_c(\boldsymbol{u})$, and $B_c(\boldsymbol{u})$ are the three color components, a digital camera aims at acquiring a discrete image $\mathcal{I}(\boldsymbol{n}) = [R(\boldsymbol{n}),\ G(\boldsymbol{n}),\ B(\boldsymbol{n})]$, with $\boldsymbol{n} \in \boldsymbol{\Gamma}$, where $\boldsymbol{\Gamma}$ is the desired sampling lattice[1].

However, existing imaging systems are subjected to some design requirements and technology limits that do not allow for the direct acquisition of the desired image. Therefore, the image acquisition process in a digital camera is more complex than the ideal sampling (see Fig. 5.1). The continuous color components $X_c(\boldsymbol{u})$, for $X = R$, G, and B, are captured at each pixel $\boldsymbol{n} \in \boldsymbol{\Gamma}$ through a spatial integration over the pixel sensor that can be represented as a convolution with a low-pass filter $p_X(\cdot)$, that is

$$X_p(\boldsymbol{n}) = p_X * X_c(\boldsymbol{n}), \quad \boldsymbol{n} \in \boldsymbol{\Gamma}. \tag{5.53}$$

[1] We assume that $\mathcal{I}_c(\boldsymbol{u})$ is bandlimited, such that sampling in the lattice $\boldsymbol{\Gamma}$ does not generate aliasing.

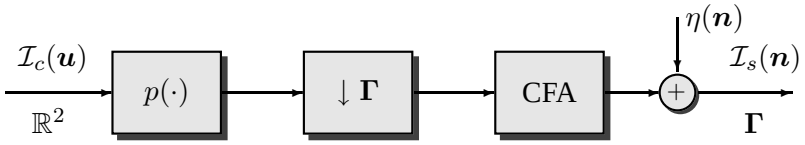


Figure 5.1: *Image acquisition in a digital camera.*

Common practical models for the impulse response of the prefilter are the gaussian or the rect functions. Moreover, a *color filter array* (CFA) usually is placed in front of the sensor, in order to acquire samples of each color component. Recently new schemes have been proposed for the CFA design, involving arrangements where the sensor pixels capture colors that can be expressed as a linear combination of the traditional red, green and blue components, as in [56, 69]. The CFA-sampled image $\mathcal{I}_s(\boldsymbol{n})$ can be expressed as

$$\mathcal{I}_s(\boldsymbol{n}) = \sum_{X=R,G,B} c_X(\boldsymbol{n}) X_p(\boldsymbol{n}) + \eta(\boldsymbol{n}), \tag{5.54}$$

where the *acquisition functions* $c_X(\boldsymbol{n})$, for $X = R$, G, and B, are periodic and, for each pixel $\boldsymbol{n} \in \boldsymbol{\Gamma}$, $c_R(\boldsymbol{n}) + c_G(\boldsymbol{n}) + c_B(\boldsymbol{n}) = 1$. Note that (5.54) describes both the classical arrangements involving only red, green and blue values, called also *pure-color*, (such as the Bayer pattern or the schemes proposed in [93]) and the new models proposed in [56, 69]. Moreover, in (5.54) the noise $\eta(\boldsymbol{n})$ introduced by the sensor during the color acquisition is considered. We assume that $\eta(\boldsymbol{n})$ is uncorrelated with respect to the acquired image $\mathcal{I}_s(\boldsymbol{n})$.

Therefore, the image acquisition process can be represented as in Fig. 5.1, and the relation between $\mathcal{I}_s(\boldsymbol{n})$, for $\boldsymbol{n} \in \boldsymbol{\Gamma}$, and the continuous image $\mathcal{I}_c(\boldsymbol{u})$ is given by

$$\mathcal{I}_s(\boldsymbol{n}) = \sum_{X=R,G,B} c_X(\boldsymbol{n}) p_X * X_c(\boldsymbol{n}) + \eta(\boldsymbol{n}), \quad \boldsymbol{n} \in \boldsymbol{\Gamma}. \tag{5.55}$$

From the acquired data $\mathcal{I}_s(\boldsymbol{n})$ the desired image $\mathcal{I}(\boldsymbol{n})$ has to be estimated. It is evident that this is an ill-posed problem due to the lost of information given by the CFA-sampling. As described in Section 5.1, a general principle for dealing with the instability of the problem is that of regularization, and the regularized solution $\hat{\mathcal{I}}$ is defined as the solution of the following problem,

$$\hat{\mathcal{I}} = \arg\min_{\mathcal{I}} \left\{ \Psi(\mathcal{I}, \mathcal{I}_s) + \sum_k \lambda_k J_k(\mathcal{I}) \right\}, \tag{5.56}$$

where the first term $\Psi(\mathcal{I}, \mathcal{I}_s)$, called *data-fidelity term*, denotes a measure of the distance between the estimated image and the observed data, while the terms $J_k(\mathcal{I})$ are some regularizing constraints based on a *a priori* knowledge of the original image. The regularization parameters λ_k control the tradeoff between the various terms.

Before describing the terms used in (5.56), let us introduce a matrix notation for the problem. Let $\boldsymbol{r}$, $\boldsymbol{g}$, and $\boldsymbol{b}$ denote the stacking vectors of the three full-resolution color components $R(\boldsymbol{n})$, $G(\boldsymbol{n})$, and $B(\boldsymbol{n})$, respectively. Let $\boldsymbol{i}$ denote the vector obtained by stacking the three color components vectors, that is $\boldsymbol{i}^T = [\boldsymbol{r}^T, \quad \boldsymbol{g}^T, \quad \boldsymbol{b}^T]$, and $\boldsymbol{i}_s$ the stacking version of the

image $\mathcal{I}_s(\boldsymbol{n})$ acquired by the sensor, while $\boldsymbol{\eta}$ is the stacking vector of the noise. Then, the relation between the acquired samples $\boldsymbol{i}_s$ and the full-resolution image $\boldsymbol{i}$ is described by the following equation,

$$\boldsymbol{i}_s = \boldsymbol{H}\boldsymbol{i} + \boldsymbol{\eta}, \tag{5.57}$$

where the matrix $\boldsymbol{H}$ is given by

$$\boldsymbol{H} = [\boldsymbol{C}_R\boldsymbol{P}_R,\ \boldsymbol{C}_G\boldsymbol{P}_G,\ \boldsymbol{C}_B\boldsymbol{P}_B], \tag{5.58}$$

with the square matrices $\boldsymbol{P}_X$ accounting for the impulse response of the filters $p_X(\boldsymbol{n})$, respectively, and the matrices $\boldsymbol{C}_X$ denoting diagonal matrices where the entries are given by stacking the acquisition functions $c_X(\boldsymbol{n})$.

Using the described matrix notation, the solution of the regularization problem in (5.56) can be found as

$$\hat{\boldsymbol{i}} = \arg\min_{\boldsymbol{i}} \left\{ \Psi(\boldsymbol{i}, \boldsymbol{i}_s) + \sum_k \lambda_k J_k(\boldsymbol{i}) \right\}. \tag{5.59}$$

The data-fidelity term $\Psi(\boldsymbol{i}, \boldsymbol{i}_s)$ can be defined according to the least-squares approaches, using the residual norm, that is

$$\Psi(\boldsymbol{i}, \boldsymbol{i}_s) = \|\boldsymbol{i}_s - \boldsymbol{H}\boldsymbol{i}\|_2^2. \tag{5.60}$$

However, a more efficient strategy consists in considering also the characteristics of the sensor noise, thus setting

$$\Psi(\boldsymbol{i}, \boldsymbol{i}_s) = \|\boldsymbol{i}_s - \boldsymbol{H}\boldsymbol{i}\|_{\boldsymbol{R}_{\boldsymbol{\eta}}^{-1}}^2 \tag{5.61}$$

where $\boldsymbol{R}_{\boldsymbol{\eta}}$ denotes the autocorrelation matrix of the noise $\boldsymbol{\eta}$.

In Sections 5.2.3 and 5.2.4 we consider two different strategies for the regularization constraints. The first one deals with quadratic penalties, allowing for efficient and fast computation methods. This solution has the drawback that it does not permit a reconstruction that adapts to the local behavior of the image. Therefore, to overcome this problem, we propose also a second approach that is spatially adaptive.

5.2.3 Quadratic Regularization

In the first approach, the regularization constraints $J_k(\boldsymbol{i})$ are chosen according to the Tikhonov method, consisting in incorporating the prior information about the image $\mathcal{I}$ through the inclusion of quadratic terms, $J_k(\boldsymbol{i}) = \|\boldsymbol{M}_k\boldsymbol{i}\|_2^2$, where $\boldsymbol{M}_k$ are appropriate matrices. This allows to find solutions with few computations.

On the basis of the characteristics of the natural color images, two constraints are considered for the solution of the demosaicking problem. The first constraint $J_1(\boldsymbol{i})$ is chosen in order to impose smoothness to each single color component, that is $\boldsymbol{M}_1 = \boldsymbol{I}_3 \otimes \boldsymbol{S}_1$, where $\boldsymbol{I}_3$ is the 3×3 identity matrix, $\otimes$ denotes the Kronecker operator and $\boldsymbol{S}_1$ represents a high-pass filter. Common choices for $\boldsymbol{S}_1$ include discrete approximations to the 2-D gradient or laplacian operators, but also two directional filters can be used, that is $\boldsymbol{S}_1 = [\boldsymbol{S}_{1h}^T, \boldsymbol{S}_{1v}^T]^T$, where $\boldsymbol{S}_{1h}$ and $\boldsymbol{S}_{1v}$ represent a horizontal and a vertical high-pass filter S_h and S_v, respectively.

The second constraint forces smoothness on the differences between the three color com-

ponents. This is the same of requiring high correlation between the high frequencies of the three color bands, which is a common assumption in many demosaicking approaches [53]. Therefore $J_2(\boldsymbol{i})$ is expressed as

$$J_2(\boldsymbol{i}) = \|\boldsymbol{S}_2\boldsymbol{r} - \boldsymbol{S}_2\boldsymbol{g}\|_2^2 + \|\boldsymbol{S}_2\boldsymbol{g} - \boldsymbol{S}_2\boldsymbol{b}\|_2^2 + \|\boldsymbol{S}_2\boldsymbol{r} - \boldsymbol{S}_2\boldsymbol{b}\|_2^2, \tag{5.62}$$

with $\boldsymbol{S}_2$ representing a high-pass filter with a cut-off frequency lower than $\boldsymbol{S}_1$, since the color differences are smoother than the single color components. In this chapter we choose the five-coefficients FIR $[0.2,\ -0.5,\ \ 0.65,\ -0.5\ \ 0.2]$ for the filters S_h and S_v, while for the second constraint the filter coefficients $[-0.5,\ 1,\ -0.5]$ are used. Equation (5.62) can be formulated also as

$$\begin{aligned} J_2(\boldsymbol{i}) &= \left\| \begin{bmatrix} \boldsymbol{S}_2 & -\boldsymbol{S}_2 & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{r} \\ \boldsymbol{g} \\ \boldsymbol{b} \end{bmatrix} \right\|_2^2 + \left\| \begin{bmatrix} \boldsymbol{0} & \boldsymbol{S}_2 & -\boldsymbol{S}_2 \end{bmatrix} \begin{bmatrix} \boldsymbol{r} \\ \boldsymbol{g} \\ \boldsymbol{b} \end{bmatrix} \right\|_2^2 \\ &\quad + \left\| \begin{bmatrix} \boldsymbol{S}_2 & \boldsymbol{0} & -\boldsymbol{S}_2 \end{bmatrix} \begin{bmatrix} \boldsymbol{r} \\ \boldsymbol{g} \\ \boldsymbol{b} \end{bmatrix} \right\|_2^2 \\ &= \boldsymbol{i}^T \begin{bmatrix} 2\boldsymbol{S}_2^T\boldsymbol{S}_2 & -\boldsymbol{S}_2^T\boldsymbol{S}_2 & -\boldsymbol{S}_2^T\boldsymbol{S}_2 \\ -\boldsymbol{S}_2^T\boldsymbol{S}_2 & 2\boldsymbol{S}_2^T\boldsymbol{S}_2 & -\boldsymbol{S}_2^T\boldsymbol{S}_2 \\ -\boldsymbol{S}_2^T\boldsymbol{S}_2 & -\boldsymbol{S}_2^T\boldsymbol{S}_2 & 2\boldsymbol{S}_2^T\boldsymbol{S}_2 \end{bmatrix} \boldsymbol{i} \\ &= \boldsymbol{i}^T \left\{ \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix} \otimes \boldsymbol{S}_2^T\boldsymbol{S}_2 \right\} \boldsymbol{i}. \end{aligned} \tag{5.63}$$

Exploiting the properties of the Kronecker product, the second regularizing constraint can be formulated on the form $J_2(\boldsymbol{i}) = \|\boldsymbol{M}_2\boldsymbol{i}\|_2^2$, with

$$\boldsymbol{M}_2 = \text{sqrt}\left(\begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix} \otimes \boldsymbol{S}_2^T\boldsymbol{S}_2 \right) = \begin{bmatrix} 1.547 & -0.577 & -0.577 \\ -0.577 & 1.547 & -0.577 \\ -0.577 & -0.577 & 1.547 \end{bmatrix} \otimes \boldsymbol{S}_2. \tag{5.64}$$

Using the two regularizing constraint $J_1(\boldsymbol{i})$ and $J_2(\boldsymbol{i})$ defined as above and the data-fidelity term defined in (5.61), the solution of (5.59) is given by solving the problem

$$\left(\boldsymbol{H}^T\boldsymbol{R}_{\boldsymbol{\eta}}^{-1}\boldsymbol{H} + \lambda_1\boldsymbol{M}_1^T\boldsymbol{M}_1 + \lambda_2\boldsymbol{M}_2^T\boldsymbol{M}_2\right)\hat{\boldsymbol{i}} - \boldsymbol{H}^T\boldsymbol{R}_{\boldsymbol{\eta}}^{-1}\boldsymbol{i}_s = 0, \tag{5.65}$$

that is $\hat{\boldsymbol{i}} = \boldsymbol{\mathcal{G}}\boldsymbol{i}_s$, with

$$\boldsymbol{\mathcal{G}} = \left(\boldsymbol{H}^T\boldsymbol{R}_{\boldsymbol{\eta}}^{-1}\boldsymbol{H} + \lambda_1\boldsymbol{M}_1^T\boldsymbol{M}_1 + \lambda_2\boldsymbol{M}_2^T\boldsymbol{M}_2\right)^{-1}\boldsymbol{H}^T\boldsymbol{R}_{\boldsymbol{\eta}}^{-1}. \tag{5.66}$$

The coefficients of the filters that estimate the three color components from the CFA sampled image can be extracted from the matrix $\boldsymbol{\mathcal{G}}$. In fact, $\boldsymbol{\mathcal{G}}$ can be written as $\boldsymbol{\mathcal{G}} = [\boldsymbol{\mathcal{G}}_R,\ \boldsymbol{\mathcal{G}}_G,\ \boldsymbol{\mathcal{G}}_B]$, where the submatrices $\boldsymbol{\mathcal{G}}_R$, $\boldsymbol{\mathcal{G}}_G$, and $\boldsymbol{\mathcal{G}}_B$ are the representation (according to the matrix notation

introduced in Section 5.2.2) of the filters that estimate the red, green and blue components from the CFA image. Due to the data sampling structure of the CFA, the resulting filter is periodically space-varying and the number of different states depends on the periodicity of the color filter array arrangement.

Estimation of the luminance

An alternative and effective demosaicking approach consists in finding an estimate $\hat{L}(\boldsymbol{n})$ of the luminance component of the image, from which the three color components can be calculated in a fast way interpolating the color differences $R - \hat{L}$, $G - \hat{L}$, and $B - \hat{L}$, as proposed in [87]. The luminance component represents the scalar intensity image containing achromatic information of the original scene, and can be expressed as a weighted positive sum of the three color channels, $L = q_R\, R + q_G\, G + q_B\, B$ [6]. In the case of the CFA-sampled image, it is suitable to define the luminance by setting each weight q_X, $X = R, G, B$, as equivalent to the ratio between the number of acquired samples for the component X and the number of pixels in the image. For example, when using the Bayer pattern, the weights $q_R = \frac{1}{4}$, $q_G = \frac{1}{2}$, and $q_B = \frac{1}{4}$ are considered.

By introducing $\boldsymbol{A} = [q_R\boldsymbol{I},\ q_G\boldsymbol{I},\ q_B\boldsymbol{I}]$, where $\boldsymbol{I}$ is the identity matrix, the stacking vector of the luminance component can be expressed as $\boldsymbol{\ell} = \boldsymbol{A}\boldsymbol{i}$. Therefore, an estimate of the luminance from the CFA-sampled data is given by

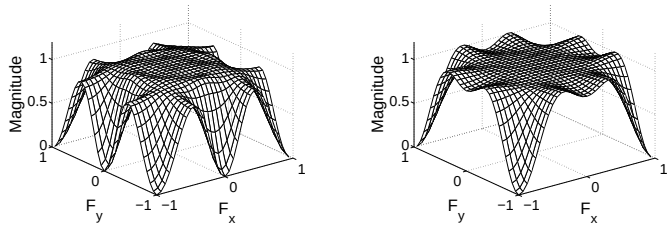
$$\hat{\boldsymbol{\ell}} = \boldsymbol{\mathcal{G}}_\ell\, \boldsymbol{i}_s, \tag{5.67}$$

where $\boldsymbol{\mathcal{G}}_\ell = \boldsymbol{A}\boldsymbol{\mathcal{G}}$ represents the filter that estimates the luminance component from the CFA data.

As for the estimation of the red, green, and blue components, also the resulting filter is periodically space-varying. As an explanatory example, consider the case, described also in [87] and in Chapter 4, of the luminance estimation when the Bayer pattern is applied, and the sensor PSFs are assumed to be an ideal impulse, $p_X(\boldsymbol{n}) = \delta(\boldsymbol{n})$. Observing the rows of the matrix $\boldsymbol{\mathcal{G}}_\ell$, two different states are noted. In the rows corresponding to the red and blue locations of the Bayer pattern the resulting filter has the frequency response as in Fig. 5.2(a). Instead, in the green pixels the frequency response of the filter results as in Fig. 5.2(b). Note that the resulting filters are in accordance with the frequency analysis carried out in Chapters 2 and 4. Since the requirements for the filter design are less demanding in the green pixels than in the red and blue positions, usually a smaller number of filter coefficients is sufficient. A similar analysis can be carried out for other CFA arrangements.

5.2.4 Spatially Adaptive Regularization

In the previous section some global properties of the images are exploited, and a general approach is applied to solve the demosaicking problem. In particular, the addition of quadratic penalties $J_k(\boldsymbol{i}) = \|\boldsymbol{M}_k\boldsymbol{i}\|_2^2$ to the least-squares (and hence quadratic) data fidelity criterion permits an efficient computation method. However, natural images are often inhomogeneous, and sharp transitions are present in correspondence of the edges. Hence, quadratic approaches



(a) (b)

Figure 5.2: *(a) Frequency response of the* 9×9 *filter used for the luminance estimation in the red/blue pixels. (b) Frequency response of the* 5×5 *filter used for the luminance estimation in the green pixels.*

produce unacceptable artifacts near the edges and the details of the image, wherein adaptive strategies are usually found to give better results.

Local adaptivity can be included in the regularization framework by considering some non-quadratic regularizing constraints of the form

$$J_k(\boldsymbol{i}) = \|\boldsymbol{M}_k \boldsymbol{i}\|^2_{\boldsymbol{W}_{\boldsymbol{i}}}, \tag{5.68}$$

where $\boldsymbol{W}_{\boldsymbol{i}}$ is a diagonal matrix estimated from the image in order to adapt the penalty term to the local features of the image [75]. If a regularization term of this type is considered together with the two quadratic penalties $J_1(\boldsymbol{i})$ and $J_2(\boldsymbol{i})$ proposed in the previous section, the solution to (5.59) is found by solving

$$\left(\boldsymbol{H}^T \boldsymbol{R}_{\boldsymbol{\eta}}^{-1} \boldsymbol{H} + \lambda_1 \boldsymbol{M}_1^T \boldsymbol{M}_1 + \lambda_2 \boldsymbol{M}_2^T \boldsymbol{M}_2 + \lambda_3 \boldsymbol{M}_3^T \boldsymbol{W}_{\boldsymbol{i}} \boldsymbol{M}_3\right) \boldsymbol{i} - \boldsymbol{H}^T \boldsymbol{R}_{\boldsymbol{\eta}}^{-1} \boldsymbol{i}_s = 0. \tag{5.69}$$

Since the matrix $\boldsymbol{W}_{\boldsymbol{i}}$ depends on $\boldsymbol{i}$, Equation (5.69) is nonlinear, and often is solved with a Landweber fixed point iteration [74, 75]. However, a large number of iterations can be required before convergence is reached, precluding fast implementations.

An alternative approach is proposed here. We use an initial estimate of the original image, denoted with $\tilde{\boldsymbol{i}}$, such that the value of $\boldsymbol{M}_3^T \boldsymbol{W}_{\tilde{\boldsymbol{i}}} \boldsymbol{M}_3 \tilde{\boldsymbol{i}}$ approximates $\boldsymbol{M}_3^T \boldsymbol{W}_{\boldsymbol{i}} \boldsymbol{M}_3 \boldsymbol{i}$. Therefore, the resulting image $\hat{\boldsymbol{i}}$ is obtained as

$$\hat{\boldsymbol{i}} = \left(\boldsymbol{H}^T \boldsymbol{R}_{\boldsymbol{\eta}}^{-1} \boldsymbol{H} + \lambda_1 \boldsymbol{M}_1^T \boldsymbol{M}_1 + \lambda_2 \boldsymbol{M}_2^T \boldsymbol{M}_2\right)^{-1} \left(\boldsymbol{H}^T \boldsymbol{R}_{\boldsymbol{\eta}}^{-1} \boldsymbol{i}_s - \lambda_3 \boldsymbol{M}_3^T \boldsymbol{W}_{\tilde{\boldsymbol{i}}} \boldsymbol{M}_3 \tilde{\boldsymbol{i}}\right). \tag{5.70}$$

Moreover, the operator $\boldsymbol{M}_3$ is set to be equivalent to the first quadratic operator $\boldsymbol{M}_1$, and both the matrices are designed using two directional filters, that is

$$\boldsymbol{M}_1 = \boldsymbol{M}_3 = \boldsymbol{I}_3 \otimes \boldsymbol{S}_1 = \boldsymbol{I}_3 \otimes \begin{bmatrix} \boldsymbol{S}_{1h} \\ \boldsymbol{S}_{1v} \end{bmatrix}, \tag{5.71}$$

in order to detect the discontinuities of the image along horizontal and vertical directions, respectively. The diagonal entries of $\boldsymbol{W}_{\tilde{\boldsymbol{i}}}$ depend on the horizontal and vertical high frequencies of the estimated image $\tilde{\boldsymbol{i}}$. In fact, $\boldsymbol{W}_{\tilde{\boldsymbol{i}}} = \text{diag}(\boldsymbol{W}_{\boldsymbol{r},h}, \ \boldsymbol{W}_{\boldsymbol{r},v}, \ \boldsymbol{W}_{\boldsymbol{g},h}, \boldsymbol{W}_{\boldsymbol{g},v}, \ \boldsymbol{W}_{\boldsymbol{b},h}, \ \boldsymbol{W}_{\boldsymbol{b},v})$, where $\text{diag}(\cdot)$ denotes the diagonal entries, and $\boldsymbol{W}_{\boldsymbol{x},h}$, $\boldsymbol{W}_{\boldsymbol{x},v}$, for $\boldsymbol{x} = \boldsymbol{r}, \boldsymbol{g}, \boldsymbol{b}$, are diagonal

matrices, which values are given by

$$\{\boldsymbol{W}_{\boldsymbol{x},h}\}_j = \xi\left(\frac{\{e_{\boldsymbol{x},v}\}_j}{\{e_{\boldsymbol{x},h}\}_j + \{e_{\boldsymbol{x},v}\}_j}\right), \tag{5.72}$$

$$\{\boldsymbol{W}_{\boldsymbol{x},v}\}_j = \xi\left(\frac{\{e_{\boldsymbol{x},h}\}_j}{\{e_{\boldsymbol{x},h}\}_j + \{e_{\boldsymbol{x},v}\}_j}\right), \tag{5.73}$$

where $\{e_{\boldsymbol{x},h}\}_j$, $\{e_{\boldsymbol{x},v}\}_j$ are the energies of the j-th value of $\boldsymbol{S}_{1h}\boldsymbol{x}$ and $\boldsymbol{S}_{1v}\boldsymbol{x}$, respectively, and $\xi(\cdot)$ is a function defined as

$$\xi(y) = \begin{cases} 0 & \text{if } y < \varepsilon \\ \dfrac{y-\varepsilon}{1-2\varepsilon} & \text{if } \varepsilon \leq y \leq 1-\varepsilon \\ 1 & \text{if } y > 1-\varepsilon \end{cases} \tag{5.74}$$

with $0 \leq \varepsilon \leq 1/2$ (in the experimental section of this chapter $\varepsilon = 0.25$ will be used). In this way, when $\{\boldsymbol{S}_{1h}\boldsymbol{x}\}_j >> \{\boldsymbol{S}_{1v}\boldsymbol{x}\}_j$, that is when the presence of a vertical edge can be assumed, $\{\boldsymbol{W}_{\boldsymbol{x},h}\}_j = 0$ and the constraint of smoothness of the color components is not considered along the horizontal direction, while it is preserved for the vertical direction. The same analysis holds when horizontal edges are found. Instead, when $\{\boldsymbol{S}_h\boldsymbol{x}\}_j$ and $\{\boldsymbol{S}_v\boldsymbol{x}\}_j$ have similar energies, smoothing is imposed along both horizontal and vertical directions.

A similar approach in designing the matrix $\boldsymbol{W}_{\boldsymbol{i}}$ was adopted by Katsaggelos *et al.* in [75] and other related works, where a *visibility function* was applied to compute the diagonal values of $\boldsymbol{W}_{\boldsymbol{i}}$. The visibility function depends on the local variance of the image and goes to zero near the edges. However, this technique does not discriminate between horizontal and vertical edges, so the high-frequency penalty is disabled for both directions. Moreover, this approach is applied in iterative restoration methods.

It can be pointed out that in (5.70) there are two smoothing penalties, since the adaptive term $J_3(\boldsymbol{i})$ is included together with the quadratic constraint $J_1(\boldsymbol{i})$. In fact $J_1(\boldsymbol{i})$ cannot be removed because the matrix $\boldsymbol{H}^T\boldsymbol{R}_{\boldsymbol{\eta}}^{-1}\boldsymbol{H} + \lambda_2\boldsymbol{M}_2^T\boldsymbol{M}_2$ is not invertible since $\ker(\boldsymbol{H}^T\boldsymbol{H}) \cap \ker(\boldsymbol{M}_2^T\boldsymbol{M}_2) \neq \{0\}$. Therefore, the regularization with respect to the spatial smoothness of the color components takes place from two constraints, where the quadratic one allows to invert the matrix $\boldsymbol{H}^T\boldsymbol{R}_{\boldsymbol{\eta}}^{-1}\boldsymbol{H} + \lambda_1\boldsymbol{M}_1^T\boldsymbol{M}_1 + \lambda_2\boldsymbol{M}_2^T\boldsymbol{M}_2$ and the second one includes adaptivity in the solution of the problem. The same approach is applied also in the half-quadratic minimization methods in the additive form [107]. However, in these approaches the diagonal submatrix for the horizontal details $\boldsymbol{W}_{\boldsymbol{x},h}$ does not consider the vertical frequencies $\boldsymbol{S}_{1v}\boldsymbol{x}$ and the computation of $\boldsymbol{W}_{\boldsymbol{x},v}$ does not take into account $\boldsymbol{S}_{1h}\boldsymbol{x}$. So local adaptivity is not based on the comparison between $\boldsymbol{S}_{1h}\boldsymbol{x}$ and $\boldsymbol{S}_{1v}\boldsymbol{x}$ as in (5.72) and (5.73), then convergence to the optimal solution is reached more slowly after many iterations.

As for the initial estimate $\tilde{\boldsymbol{i}}$ used in the equation (5.70), an efficient solution is to apply the quadratic approach described in Section 5.2.3, that is $\tilde{\boldsymbol{i}} = \boldsymbol{\mathcal{G}}\boldsymbol{i}_s$. In this way the approximation $\boldsymbol{M}_3^T\boldsymbol{W}_{\tilde{\boldsymbol{i}}}\boldsymbol{M}_3\tilde{\boldsymbol{i}} \simeq \boldsymbol{M}_3^T\boldsymbol{W}_{\boldsymbol{i}}\boldsymbol{M}_3\boldsymbol{i}$ is verified, and the proposed scheme provides a reliable estimate of the color image $\boldsymbol{i}$, as proved by the experimental results reported in Section 5.2.6.

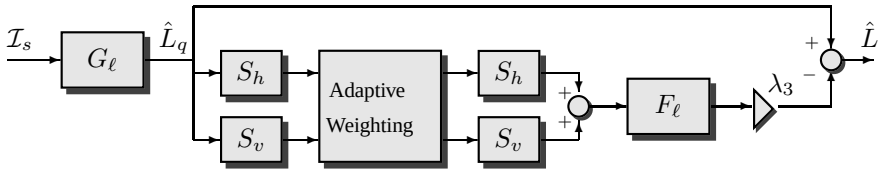


Figure 5.3: *Scheme of the adaptive estimation of the luminance.*

Estimation of the luminance with the adaptive scheme

As described in Section 5.2.3, the estimation of the luminance is preferred, instead of the three color components, in order to reduce the computational cost of the algorithm. In fact, red, green and blue values can be computed through bilinear interpolation of the color differences using the estimated luminance as a reference component.

Considering that the luminance can be expressed as $\boldsymbol{\ell} = \boldsymbol{A}\boldsymbol{i}$, as seen above, an estimate of the luminance with the spatially adaptive scheme of Equation (5.70) results

$$\hat{\boldsymbol{\ell}} = \boldsymbol{A}\left(\boldsymbol{H}^T\boldsymbol{R}_{\boldsymbol{\eta}}^{-1}\boldsymbol{H} + \lambda_1\boldsymbol{M}_1^T\boldsymbol{M}_1 + \lambda_2\boldsymbol{M}_2^T\boldsymbol{M}_2\right)^{-1}\left(\boldsymbol{H}^T\boldsymbol{R}_{\boldsymbol{\eta}}^{-1}\boldsymbol{i}_s - \lambda_3\boldsymbol{M}_3^T\boldsymbol{W}_{\tilde{\boldsymbol{i}}}\boldsymbol{M}_3\tilde{\boldsymbol{i}}\right). \quad (5.75)$$

Since $\boldsymbol{M}_3 = \boldsymbol{I}_3 \otimes \boldsymbol{S}_1$ and $\boldsymbol{W}_{\tilde{\boldsymbol{i}}} = \text{diag}\left(\boldsymbol{W}_{\tilde{\boldsymbol{r}}}, \boldsymbol{W}_{\tilde{\boldsymbol{g}}}, \boldsymbol{W}_{\tilde{\boldsymbol{b}}}\right)$, with $\boldsymbol{W}_{\tilde{\boldsymbol{x}}} = \text{diag}\left(\boldsymbol{W}_{\tilde{\boldsymbol{x}},h}, \boldsymbol{W}_{\tilde{\boldsymbol{x}},v}\right)$, we have

$$\boldsymbol{M}_3^T\boldsymbol{W}_{\tilde{\boldsymbol{i}}}\boldsymbol{M}_3\tilde{\boldsymbol{i}} = \begin{bmatrix} \boldsymbol{S}_1^T\boldsymbol{W}_{\tilde{\boldsymbol{r}}}\boldsymbol{S}_1\tilde{\boldsymbol{r}} \\ \boldsymbol{S}_1^T\boldsymbol{W}_{\tilde{\boldsymbol{g}}}\boldsymbol{S}_1\tilde{\boldsymbol{g}} \\ \boldsymbol{S}_1^T\boldsymbol{W}_{\tilde{\boldsymbol{b}}}\boldsymbol{S}_1\tilde{\boldsymbol{b}} \end{bmatrix}, \quad (5.76)$$

and, considering that the high frequencies of the three color components are highly correlated with those of the luminance, the following approximation holds

$$\boldsymbol{S}_1^T\boldsymbol{W}_{\tilde{\boldsymbol{r}}}\boldsymbol{S}_1\tilde{\boldsymbol{r}} \simeq \boldsymbol{S}_1^T\boldsymbol{W}_{\tilde{\boldsymbol{g}}}\boldsymbol{S}_1\tilde{\boldsymbol{g}} \simeq \boldsymbol{S}_1^T\boldsymbol{W}_{\tilde{\boldsymbol{b}}}\boldsymbol{S}_1\tilde{\boldsymbol{b}} \simeq \boldsymbol{S}_1^T\boldsymbol{W}_{\tilde{\boldsymbol{\ell}}}\boldsymbol{S}_1\tilde{\boldsymbol{\ell}}. \quad (5.77)$$

Therefore, term in (5.76) can be replaced with

$$\boldsymbol{M}_3^T\boldsymbol{W}_{\tilde{\boldsymbol{i}}}\boldsymbol{M}_3\tilde{\boldsymbol{i}} = \boldsymbol{I}_3 \otimes \boldsymbol{S}_1^T\boldsymbol{W}_{\tilde{\boldsymbol{\ell}}}\boldsymbol{S}_1\tilde{\boldsymbol{\ell}}. \quad (5.78)$$

By introducing the matrix $\boldsymbol{F}_\ell = \boldsymbol{A}\left(\boldsymbol{H}^T\boldsymbol{R}_{\boldsymbol{\eta}}^{-1}\boldsymbol{H} + \lambda_1\boldsymbol{M}_1^T\boldsymbol{M}_1 + \lambda_2\boldsymbol{M}_2^T\boldsymbol{M}_2\right)^{-1}[\boldsymbol{I}, \boldsymbol{I}, \boldsymbol{I}]^T$, equation (5.75) becomes

$$\hat{\boldsymbol{\ell}} = \boldsymbol{\mathcal{G}}_\ell\boldsymbol{i}_s - \lambda_3\boldsymbol{F}_\ell\boldsymbol{S}_1^T\boldsymbol{W}_{\tilde{\boldsymbol{\ell}}}\boldsymbol{S}_1\tilde{\boldsymbol{\ell}} \quad (5.79)$$

where $\boldsymbol{\mathcal{G}}_\ell$ is defined in the previous section. If the initial estimate of the luminance $\tilde{\boldsymbol{\ell}}$ is computed with the quadratic approach described in Section 5.2.3, that is $\tilde{\boldsymbol{\ell}} = \boldsymbol{\mathcal{G}}_\ell\boldsymbol{i}_s$, equation (5.79) can be written as

$$\hat{\boldsymbol{\ell}} = \left(\boldsymbol{I} - \lambda_3\boldsymbol{F}_\ell\boldsymbol{S}_1^T\boldsymbol{W}_{\tilde{\boldsymbol{\ell}}}\boldsymbol{S}_1\right)\boldsymbol{\mathcal{G}}_\ell\boldsymbol{i}_s. \quad (5.80)$$

Equation (5.80) indicates the procedure to compute the luminance from the CFA-sampled image with the proposed adaptive method. The resulting scheme is reported in Figure 5.3, where G_ℓ is the space-varying filter designed in Section 5.2.3 and the filter F_ℓ is obtained by matrix

$\boldsymbol{F}_\ell$. Filters S_h and S_v are the horizontal and vertical high-pass filters represented by matrices $\boldsymbol{S}_{1h}$ and $\boldsymbol{S}_{1v}$, respectively.[2]

5.2.5 Determination of the regularization parameters

The regularization parameters λ_k have to be chosen in order to give the best tradeoff between the regularizing constraints and the data fidelity term $\Psi(\boldsymbol{i}, \boldsymbol{i}_s)$. Moreover, the determination of these parameters turns out to be critical due to the presence of multiple constraints. In fact, also the relative values between them are of vital importance for the quality of the estimation. In literature, this has been the subject of extensive research [13, 74] and several methods for choosing the regularization parameters have been discussed.

Initially, we concentrate our analysis on the parameters λ_1 and λ_2 for the quadratic method presented in Section 5.2.3. Using the approach proposed in [13], based on a multiparameter generalization of the L-curve method, it is found that the optimal solution is reached when $\lambda_2 \simeq 20\lambda_1$. This knowledge is enough when the noise $\boldsymbol{\eta}$ in (5.57) is small and has a limited energy, but in the general case also the actual values of λ_1 and λ_2 are needed. To determine λ_1 and λ_2 we tested the proposed approach on 20 images of the Kodak data-set, reported also in [52], adding a white gaussian noise, which variance σ_η^2 was varied from 0 to 225. We found that for all values of σ_η^2 the best reconstruction minimizing the mean square error is achieved by $\lambda_1 = 0.001$ and $\lambda_2 = 0.02$.

For the adaptive strategy we prefer to choose the parameters using a subjective visual approach. In fact, the regularization parameter λ_3 plays an important role because it affects the influence of the adaptive constraint on the regularization of the solution, and its effects are noticeable especially near the edges and the details of the image. When the Bayer pattern is used, it is found that good performances are given by setting $\lambda_1 = 0.0008$, $\lambda_2 = 0.02$, and $\lambda_3 = 0.0016$.

5.2.6 Experimental Results

In this section we present some experimental results to evaluate the performance of the proposed regularization approaches. In particular, we analyze the performances of the proposed algorithms for the reconstruction of images acquired in five different situations:

1. using the Bayer pattern;
2. using the CFA arrangement proposed by Lukac [93], that employs RGB sensors;
3. using the CFA pattern proposed by Hirakawa and Wolfe [69] involving new panchromatic sensors;
4. using the Bayer pattern and assuming that the CFA data are corrupted by a gaussian noise;
5. using the Bayer pattern and sensors with a non ideal impulse response.

[2] In Figure 5.3 is assumed that filters S_h and S_v are even-symmetric because in this case $\boldsymbol{S}_{1h}^T = \boldsymbol{S}_{1h}$ and $\boldsymbol{S}_{1v}^T = \boldsymbol{S}_{1v}$. Instead, if S_h and S_v are odd-symmetric $\boldsymbol{S}_{1h}^T = -\boldsymbol{S}_{1h}$ and $\boldsymbol{S}_{1v}^T = -\boldsymbol{S}_{1v}$, after the adaptive weighting S_h and S_v have to be replaced by $-S_h$ and $-S_v$, respectively.

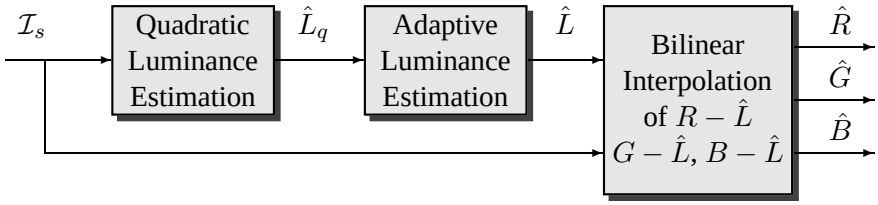


Figure 5.4: *Complete scheme of the proposed adaptive color reconstruction.*

Experiment 1: Demosaicking of images sampled with the Bayer pattern

In the first experiment we use the proposed approaches for the reconstruction of images sampled with the Bayer pattern (see Fig. 2.1(a)), the most popular CFA arrangement, for which many algorithms have been presented in the literature.

Both quadratic and adaptive regularization methods are considered. In the first one, the luminance component is estimated with the approach described in Section 5.2.3, and the full color representation of the image is reconstructed through bilinear interpolation of the color differences $R - \hat{L}$, $G - \hat{L}$ and $B - \hat{L}$ as in [87]. On the other one, the adaptive estimate of the luminance is obtained with the procedure sketched in Fig. 5.3 in the red and blue pixels only, since in the green pixels the quadratic regularization is sufficient, because the requirements for the luminance estimation allow a simpler design of the filter, as shown in Section 5.2.3. Finally, bilinear interpolation is performed to obtain the three color component values. The complete scheme of the proposed adaptive reconstruction from the CFA samples is reported in Fig. 5.4. The regularization parameters are chosen as described in Section 5.2.5, that is $\lambda_1 = 0.0008$, $\lambda_2 = 0.02$ and $\lambda_3 = 0.0016$. The periodically space-varying filter G_ℓ has dimension 5×5 in the green pixels, while the filters G_ℓ and F_ℓ in the red and blue location are 9×9.

In Table 5.2 the resulting CPSNR values are compared with the values given by the algorithms proposed in [p4, 6, 52, 64, 83, 87, 143] (using the original MATLAB code provided by the authors). Numbers in boldface indicate the highest values. The quadratic approach performs better than other non-adaptive methods such as the technique of [6], but fails to give good results for the most detailed images. Instead, the adaptive reconstruction is able to work properly also in non-homogeneous regions of the image and provides good numerical results. In most cases, the performances are better or at least comparable with the ones obtained using more computational demanding algorithms. A visual inspection of the reconstructed images confirms the soundness of the approach, since also the finest details are preserved, as shown in Fig. 5.5, where a particular of the test image #6 is reconstructed using the algorithms presented in [87, 143] and the proposed adaptive technique.

Experiment 2: Demosaicking of images acquired with other CFA arrangements

Since many works are recently focused on different design and geometries for the sensor, it is interesting to consider new demosaicking approaches giving good performances not only with the Bayer pattern, but with any lattice structure. Two algorithms that operate independently from the color arrangements are proposed in [22, 94] for CFA involving RGB sensors. The regularization methods described in this chapter can be applied to any type of the CFA. To give

Figure 5.5: *Portion of the image no. 6 of the Kodak set: (a) original image; (b) image reconstructed by the technique presented [143]; (c) image reconstructed by the technique presented [87]; (d) image reconstructed by the proposed adaptive approach.*

a demonstration of this characteristic, we tested the regularization approaches with the pattern proposed by Lukac [93] (see Fig. 2.1(b)). The full color images are reconstructed with the quadratic approach (using the parameter values described in Section 5.2.5), and according to the adaptive scheme of Fig. 5.4, setting $\lambda_1 = 0.0013$, $\lambda_2 = 0.02$, and $\lambda_3 = 0.0019$. All the filters have dimension 7×7.

In Table 5.3 the performances of the regularization methods are compared with the results obtained by the algorithms of [22, 94]. For the algorithms of [22] the MATLAB code provided by the authors is used. The CPSNR values given by the quadratic and the adaptive approaches are higher than the values provided by the other methods. The images reconstructed with the proposed techniques show significant visual quality improvements with respect to the other approaches. An example is given in Fig. 5.6 for a portion of the test image *lighthouse*.

Experiment 3: Demosaicking of images acquired with CFA arrangements involving panchromatic sensors

As described in Sections 2.1 and 5.2.2, new CFA arrangements apply sensors that do not acquire color in the RGB space such as the Bayer pattern, but the color components can be expressed as a linear combination of the red, green and blue values. Several classical demosaicking approaches proposed for the Bayer pattern cannot be extended to these new configurations and require a change of basis. Hirakawa and Wolfe describe some of these new arrangements in [69] and propose a demosaicking procedure to reconstruct images acquired with these configurations in [68]. Table 5.4 compares the performances of this approach with the algorithms

Method	[p4]	[6]	[52]	[64]	[83]	[87]	[143]	Quadratic Approach	Adaptive Approach
1	36.91	36.18	37.90	35.15	38.44	37.58	**38.36**	36.79	38.04
2	40.16	36.16	38.54	38.88	35.51	40.26	**40.42**	37.32	38.16
3	37.44	35.62	37.23	35.48	35.08	38.12	38.05	36.83	**38.17**
4	39.24	37.08	38.68	37.59	39.14	38.11	**40.03**	37.43	39.66
5	41.61	38.89	41.21	40.33	39.11	**42.69**	42.03	40.68	42.39
6	35.42	32.52	35.56	33.83	35.66	35.37	35.96	33.27	**36.01**
7	42.34	39.37	41.71	40.96	41.46	42.72	**42.76**	40.69	42.38
8	42.14	39.67	41.06	40.56	40.90	**42.69**	41.77	41.12	42.46
9	39.30	37.47	39.34	37.59	38.91	39.47	**40.10**	38.51	39.78
10	42.95	39.24	41.86	41.57	41.65	42.77	42.81	40.82	**43.07**
11	33.33	34.52	34.46	31.45	**35.71**	33.89	34.73	34.77	34.62
12	39.01	36.54	38.40	37.72	37.25	**39.53**	39.17	37.91	39.18
13	43.05	39.82	41.84	41.49	42.14	41.30	43.26	40.63	**43.28**
14	40.77	39.95	40.93	39.31	41.12	41.58	41.43	41.06	**41.59**
15	36.10	35.63	36.50	34.12	36.21	36.76	36.48	36.82	**37.01**
16	39.82	36.51	39.80	38.04	40.16	40.01	**40.46**	37.88	39.97
17	39.87	38.23	37.48	38.55	39.19	40.54	37.80	39.04	**40.63**
18	37.93	37.25	38.67	36.50	39.13	38.70	38.93	38.15	**39.16**
19	37.74	36.50	37.73	36.35	37.17	38.64	**38.55**	37.56	38.47
20	34.21	34.36	34.14	32.79	34.54	34.64	34.91	35.09	**35.30**
Ave.	38.97	37.08	38.65	37.41	38.42	39.27	39.40	38.12	**39.47**

Table 5.2: *CPSNR* (dB) *for different demosaicking methods using the Bayer pattern.*

Method	[22]	[94]	Quadratic Approach	Adaptive Approach
1	34.05	30.78	36.54	**37.58**
2	37.82	36.57	37.25	**38.49**
3	32.25	30.93	35.77	**36.79**
4	35.22	32.02	37.08	**38.95**
5	38.19	37.09	40.08	**41.63**
6	32.27	28.67	34.86	**35.97**
7	39.04	36.64	40.36	**41.74**
8	38.98	36.80	40.77	**41.97**
9	36.06	33.78	38.31	**39.32**
10	39.58	37.11	40.33	**42.44**
11	30.55	28.55	**34.27**	33.96
12	37.10	35.81	37.94	**38.87**
13	38.90	35.46	40.25	**42.59**
14	37.60	36.37	40.55	**40.83**
15	33.40	31.73	36.06	**36.16**
16	35.53	32.51	37.73	**39.25**
17	37.07	32.93	38.31	**39.42**
18	35.40	32.83	37.74	**38.63**
19	36.01	34.35	37.21	**37.83**
20	32.56	30.32	34.81	**34.91**
Ave.	35.88	33.56	37.81	**38.87**

Table 5.3: *CPSNR* (dB) *for different demosaicking methods using the CFA arrangement proposed by Lukac.*

(a) (b) (c)

Figure 5.6: *Portion of the image no. 16 of the Kodak set: (a) original image; (b) image sampled with the pattern of Fig. 2.1(b) and reconstructed by the technique presented in [22]; (c) image sampled with the pattern of Fig. 2.1(b) and reconstructed by the proposed adaptive approach.*

presented in Sections 5.2.3 and 5.2.4, respectively, after acquisition with the arrangement of Fig. 2.1(d). The regularization parameters are $\lambda_1 = 0.0013$, $\lambda_2 = 0.02$, and $\lambda_3 = 0.0008$, and the filters have dimension 9×9. We observe that the proposed methods improve the quality of the reconstruction images obtaining a higher CPSNR average value.

Experiment 4: Demosaicking of noisy images

As seen in Section 5.2.2 the acquisition of an image from the sensor introduces a noise component. To see the robustness of the different demosaicking approaches to the noise effects, we test the same algorithms used in the Experiment 1 when the data captured with a CFA arranged according to the Bayer pattern are corrupted by a gaussian noise η with standard deviation $\sigma_\eta = 5$. The performances are reported in Table 5.5. As for the proposed methods, the same parameter values considered in the Experiment 1 are used. The CPSNR values are lower than the results obtained in the noise-free case reported in Table 5.2. In particular, the noise makes difficult the edge estimation performed by the adaptive approaches, so the quadratic strategy outperforms many demosaicking strategies such as [p4, 52] and [83], despite the reduced computational cost. The adaptive technique described in Section 5.2.4 provides a gain of 0.27 dB with respect to the method proposed in Section 5.2.3 and gives the higher CPSNR average value. Afterwards, the quality of the images can be improved applying a denoising procedure. An alternative strategy could be to use techniques that perform demosaicking jointly to denoising, such as the method proposed in [144].

Experiment 5: Demosaicking of images acquired with non-impulsive sensors

The quality of the demosaicked images strongly depends on the characteristics of the sensors. In fact, sensors with a bandlimited frequency response avoid aliasing and make easier the demosaicking procedure, but produce smoothed images. Therefore, several digital cameras apply a deblurring algorithm to sharpen the demosaicked image. The techniques presented in this chapter can overcome this problem. In fact, the design of the filters used for the color recon-

Method	[68]	Quadratic Approach	Adaptive Approach
1	38.97	**39.71**	39.42
2	**38.96**	37.96	38.30
3	36.76	36.68	**37.04**
4	40.13	**40.74**	40.58
5	**41.40**	40.65	41.27
6	36.87	37.22	**37.23**
7	**42.10**	41.73	42.05
8	42.21	42.11	**42.36**
9	39.82	39.98	**40.04**
10	43.00	42.76	**43.07**
11	34.58	**36.04**	35.37
12	**39.42**	38.91	39.20
13	44.00	**44.41**	44.27
14	41.11	**41.76**	41.58
15	36.77	**36.78**	36.77
16	**40.63**	40.61	40.62
17	**40.47**	40.16	40.36
18	39.77	**39.80**	39.71
19	**38.73**	38.33	38.56
20	35.91	**36.16**	36.10
Ave.	39.58	39.63	**39.70**

Table 5.4: *CPSNR* (dB) *for different demosaicking methods using one of the CFA arrangements proposed by Hirakawa and Wolfe.*

Method	[p4]	[6]	[52]	[64]	[83]	[87]	[143]	Quadratic Approach	Adaptive Approach
1	32.26	32.54	32.63	31.70	32.86	32.91	**33.06**	32.58	32.95
2	33.13	32.55	32.84	33.08	31.52	**33.67**	33.62	33.29	33.42
3	32.48	32.30	32.46	31.89	31.39	33.15	33.03	32.86	**33.17**
4	33.02	32.98	32.96	32.73	33.11	33.21	33.54	32.98	**33.59**
5	33.44	33.52	33.43	33.40	33.12	34.20	33.97	34.17	**34.33**
6	31.73	30.60	31.81	31.09	31.82	32.00	**32.19**	30.76	32.06
7	33.53	33.64	33.51	33.52	33.46	34.18	34.07	34.09	**34.30**
8	33.52	33.73	33.41	33.47	33.42	34.21	33.92	34.20	**34.33**
9	32.96	33.07	33.03	32.64	32.89	33.54	33.52	33.41	**33.63**
10	33.67	33.64	33.57	33.64	33.54	34.24	34.07	34.13	**34.40**
11	30.71	31.77	31.35	29.67	**31.97**	31.33	31.71	31.38	31.29
12	33.13	32.79	33.01	32.89	32.45	33.70	33.48	33.46	**33.74**
13	33.57	33.76	33.49	33.60	33.68	33.93	34.05	33.97	**34.29**
14	33.33	33.81	33.42	33.24	33.55	34.03	33.89	34.12	**34.16**
15	32.02	32.31	32.23	31.28	32.18	32.71	32.52	**32.78**	32.73
16	33.12	32.69	33.12	32.84	33.31	33.63	33.62	33.14	**33.69**
17	33.96	33.88	33.24	33.63	33.78	34.49	33.53	34.29	**34.68**
18	32.64	32.98	32.92	32.32	33.05	33.37	33.31	33.24	**33.46**
19	32.58	32.70	32.62	32.31	32.36	33.35	33.19	33.24	**33.41**
20	31.23	31.71	31.22	30.68	31.48	31.80	31.85	32.03	**32.05**
Ave.	32.80	32.85	32.81	32.48	32.75	33.38	33.31	33.21	**33.48**

Table 5.5: *CPSNR (*dB*) for different demosaicking methods using the Bayer pattern and data corrupted by a gaussian noise with* $\sigma_\eta = 5$.

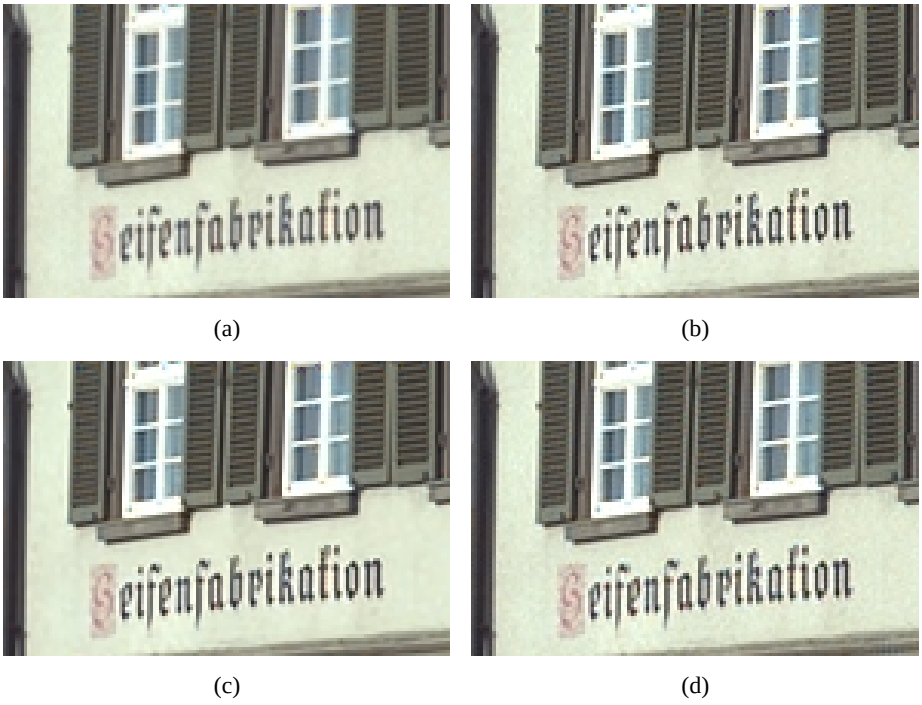


Figure 5.7: *Portion of the image no. 6 of the Kodak set: (a) image reconstructed by the technique presented in [143] (without deblurring); (b) image reconstructed by the technique presented in [143] and sharpened with the method proposed in [15]; (c) image reconstructed by the technique presented in [143] and sharpened with the method proposed in [81]; (d) image reconstructed by the proposed adaptive approach.*

struction takes into account the impulse response of the sensors (see Section 5.2.2), adapting demosaicking to its characteristics and performing deblurring jointly.

Table 5.6 reports the performances of several approaches when the sensor pixels have a gaussian impulse response with standard deviation $\sigma_s = 0.5$. The CFA is arranged with the Bayer pattern. The demosaicking techniques proposed in [87, 143] followed by various deblurring algorithms are considered for comparison. In particular the total variation-based image deconvolution proposed in [15], and the deconvolution using a sparse prior [81] are considered[3]. Their performances are compared with the results obtained with the proposed regularization methods. The regularization parameters and the filter dimensions are set as in the first experiment.

The quality of the demosaicked images provided by the algorithms in [87] and [143] without deblurring is poor, while a restoration algorithm after the color reconstruction improves considerably the performances, providing average CPSNR improvements up to 3.6 dB. The proposed adaptive strategy is able to produce sharpened demosaicked images, so the use of enhancement procedures is not necessary. The CPSNR average value obtained by the proposed method is higher than those given by the demosaicking techniques [87, 143] followed by com-

[3]The MATLAB code made available by the authors is used for both the deblurring algorithms [15] and [81]. It can be downloaded at `http://www.lx.it.pt/~bioucas`, and at `http://groups.csail.mit.edu/graphics/CodedAperture/DeconvolutionCode.html`, respectively.

Demosaicking	[87]			[143]			Quadratic	Adaptive
Deblurring		[15]	[81]		[15]	[81]	Approach	Approach
1	33.20	38.13	38.29	33.47	38.26	**38.85**	37.03	38.07
2	37.98	39.74	39.77	38.06	**40.42**	39.78	37.86	38.24
3	33.25	37.24	37.11	33.24	38.08	37.05	36.98	**38.16**
4	34.35	38.57	38.60	34.81	**40.03**	39.83	37.58	39.81
5	39.29	41.36	41.50	39.08	42.03	40.77	41.20	**42.71**
6	30.76	35.77	35.84	30.97	35.90	36.19	33.66	**36.32**
7	39.07	42.33	42.32	39.16	**42.76**	42.04	41.02	42.59
8	38.77	42.22	42.20	38.50	41.77	41.06	41.38	**42.60**
9	35.63	39.38	39.40	35.81	**40.10**	39.74	38.56	39.78
10	39.39	42.65	42.70	39.41	42.81	42.03	41.33	**43.35**
11	30.06	34.21	34.47	30.29	34.59	**35.06**	34.28	34.25
12	37.05	39.13	39.17	36.87	39.17	38.41	38.28	**39.37**
13	37.87	41.88	41.91	38.37	43.26	42.95	40.79	**43.50**
14	37.98	41.36	41.35	37.95	41.43	40.98	41.01	**41.54**
15	33.62	**36.72**	36.54	33.54	36.45	36.13	36.55	**36.72**
16	35.31	39.98	40.05	35.49	**40.46**	40.23	38.14	40.33
17	37.15	40.54	40.40	35.89	37.80	37.43	39.11	**40.72**
18	35.00	38.69	38.92	35.10	38.93	39.07	38.20	**39.12**
19	36.06	**38.61**	38.11	36.02	38.56	37.77	37.69	38.53
20	32.08	34.63	34.70	32.14	34.91	34.59	34.90	**35.15**
Ave.	35.69	39.16	39.17	35.71	39.39	39.00	38.28	**39.54**

Table 5.6: *CPSNR* (dB) *for different demosaicking methods using the Bayer pattern and sensors with a gaussian impulse response with* $\sigma_s = 0.5$.

putational demanding deblurring methods such as [15, 81]. A visual comparison is given in Fig. 5.7 (the original image is shown in Fig. 5.5(a)). The image shown in Fig. 5.7(a), reconstructed without sharpening procedures, is blurred, while the two deblurred samples present different drawbacks. In Fig 5.7(b) the deblurring algorithm amplifies the demosaicking artifacts, while Fig 5.7(c) is too smoothed in the homogeneous regions. The best compromise between sharpness and absence of demosaicking artifacts is given by the image reconstructed with the proposed adaptive approach, shown in the last figure.

Chapter 6

Joint Demosaicking and Denoising

In a digital camera usually a denoising procedure is required to remove the noise introduced by the sensor. In this chapter we analyze demosaicking and denoising jointly. The color reconstruction is performed using space-varying filters, therefore a formulation for the evaluation of the autocorrelation of the noise filtered with a space-varying filter is reported. This approach gives good performances and the computational cost is kept low.

6.1 Introduction

As described in Chapters 1 and 2, a digital camera pipeline needs a denoising procedure to remove the noise introduced by the sensor. Moreover, demosaicking and denoising cannot be considered separately since each process affects the other. In fact, to apply denoising before demosaicking is sub-optimal due to the particular structure of the CFA image, while performing denoising on the demosaicked image becomes difficult because demosaicking can considerably modify the statistics of the noise with respect to the noise introduced by the sensor [73]. In particular, many demosaicking methods apply periodical space-varying filters and this invalidates the assumption of stationarity of the noise, which is the basis of many denoising algorithms. Moreover, denoising of the color reconstructed images requires to denoise the three color components, thus increasing the computational cost.

Therefore, several approach to perform demosaicking and denoising jointly have been presented [62, 63, 65, 108, 144] (see Section 2.9 for a brief introduction about them).

In this chapter, we propose to solve this problem extending the demosaicking method introduced in Chapter 5. The computational cost is kept low and the same approach can be applied to reconstruct images acquired with any type of CFA arrangement.

Since the noise is usually assumed white gaussian and the technique presented in Chapter 5 is based on linear periodical space-varying (LPSV) filters (as other demosaicking approaches), in the next section we analyze the characteristics of a gaussian random process filtered with a LPSV filter. On the basis of the results of this analysis, in Section 6.3 we describe a strategy to remove the noise. In Section 6.4 an efficient algorithm to perform joint demosaicking and denoising with CFA arrangements involving RGB sensors is proposed. Finally, in Section 6.5 we report the experimental results of the proposed approach.

6.2 Random processes filtered with LPSV filters

A linear periodical space-varying (LPSV) system is characterized by the following input-output relation

$$\eta_X(\boldsymbol{n}) = \sum_{\boldsymbol{m}} p(\boldsymbol{n}, \boldsymbol{m})\eta(\boldsymbol{n} - \boldsymbol{m}) \tag{6.1}$$

where the impulse response is periodic, that is $p(\boldsymbol{n}, \boldsymbol{m}) = p(\boldsymbol{n} + \boldsymbol{r}, \boldsymbol{m})$, $\forall \boldsymbol{r} \in \Lambda(\boldsymbol{V})$, with $\Lambda(\boldsymbol{V})$ a lattice formed by columns of $\boldsymbol{V}$, and $\boldsymbol{V}$ is called *periodicity matrix* [38]. Therefore $p(\boldsymbol{n}, \boldsymbol{m})$ admits a Fourier representation given by

$$p(\boldsymbol{n}, \boldsymbol{m}) = \sum_{\boldsymbol{k} \in \mathcal{I}_{\boldsymbol{V}}} p_{\boldsymbol{k}}(\boldsymbol{m}) e^{j2\pi \boldsymbol{k}^T \boldsymbol{V}^{-1} \boldsymbol{n}}, \tag{6.2}$$

where the $p_{\boldsymbol{k}}(\boldsymbol{m})$ are the polyphase components and $\mathcal{I}_{\boldsymbol{V}}$ denotes a finite-extent region that has at most $|\det(\boldsymbol{V})|$ elements. Thus, the input-output relation (6.1) can be expressed as

$$\eta_X(\boldsymbol{n}) = \sum_{\boldsymbol{k}} e^{j2\pi \boldsymbol{k}^T \boldsymbol{V}^{-1} \boldsymbol{n}} \eta_{\boldsymbol{k}}(\boldsymbol{n}) \tag{6.3}$$

where $\eta_{\boldsymbol{k}}(\boldsymbol{n}) = p_{\boldsymbol{k}} * \eta(\boldsymbol{n})$, with $*$ denoting the 2D-convolution operator. If the input $\eta(\boldsymbol{n})$ is a random process, the autocorrelation function of the output process $\eta_X(\boldsymbol{n})$ can be expressed as

$$\begin{aligned} r_{\eta_X}(\boldsymbol{n}, \boldsymbol{m}) &= E[\eta_X(\boldsymbol{n})\eta_X(\boldsymbol{n} + \boldsymbol{m})] \\ &= \sum_{\boldsymbol{k}} \sum_{\boldsymbol{h}} e^{j2\pi \boldsymbol{k}^T \boldsymbol{V}^{-1} \boldsymbol{n}} e^{j2\pi \boldsymbol{h}^T \boldsymbol{V}^{-1} (\boldsymbol{n} + \boldsymbol{m})} E[\eta_{\boldsymbol{k}}(\boldsymbol{n})\eta_{\boldsymbol{h}}(\boldsymbol{n} + \boldsymbol{m})]. \end{aligned} \tag{6.4}$$

In particular, if $\eta(\boldsymbol{n})$ is white gaussian, $\eta(\boldsymbol{n}) \sim \mathcal{N}(0, \sigma^2)$, the autocorrelation function of $\eta_G(\boldsymbol{n})$ is

$$r_{\eta_X}(\boldsymbol{n}, \boldsymbol{m}) = \sigma^2 \sum_{\boldsymbol{k}} \sum_{\boldsymbol{h}} e^{j2\pi \boldsymbol{k}^T \boldsymbol{V}^{-1} \boldsymbol{n}} e^{j2\pi \boldsymbol{h}^T \boldsymbol{V}^{-1} (\boldsymbol{n} + \boldsymbol{m})} p_{\boldsymbol{k}} * p_{\boldsymbol{h}}(\boldsymbol{m}), \tag{6.5}$$

assuming that all the filters $p_{\boldsymbol{k}}(\boldsymbol{m})$ are symmetric.

Therefore it is evident that the output process after a space-varying filter is non stationary.

Now we consider the system shown in Fig. 6.1, that is when a space-invariant filter $w(\cdot)$ is applied after a LPSV filter. The input process $\eta(\boldsymbol{n})$ is assumed white gaussian, therefore the autocorrelation of $\eta_X(\boldsymbol{n})$ is given by (6.5).

Then, the autocorrelation of $\eta_W(\boldsymbol{n})$ after the filter $w(\cdot)$ is given by

$$\begin{aligned} r_{\eta_W}(\boldsymbol{n}, \boldsymbol{m}) &= \sum_{\boldsymbol{\alpha}} \sum_{\boldsymbol{\beta}} w(\boldsymbol{\alpha}) w(\boldsymbol{\beta}) E[\eta_X(\boldsymbol{n} - \boldsymbol{\alpha})\eta_X(\boldsymbol{n} + \boldsymbol{m} - \boldsymbol{\beta})] \\ &= \sum_{\boldsymbol{\alpha}} \sum_{\boldsymbol{\beta}} w(\boldsymbol{\alpha}) w(\boldsymbol{\beta}) r_{\eta_X}(\boldsymbol{n} - \boldsymbol{\alpha}, \boldsymbol{m} - \boldsymbol{\beta} + \boldsymbol{\alpha}) \\ &= \sigma^2 \sum_{\boldsymbol{\alpha}} \sum_{\boldsymbol{\beta}} w(\boldsymbol{\alpha}) w(\boldsymbol{\beta}) \sum_{\boldsymbol{k}} \sum_{\boldsymbol{h}} e^{j2\pi \boldsymbol{k}^T \boldsymbol{V}^{-1} (\boldsymbol{n} - \boldsymbol{\alpha})} e^{j2\pi \boldsymbol{h}^T \boldsymbol{V}^{-1} (\boldsymbol{n} + \boldsymbol{m} - \boldsymbol{\beta})} \\ &\quad p_{\boldsymbol{k}} * p_{\boldsymbol{h}}(\boldsymbol{m} - \boldsymbol{\beta} - \boldsymbol{\alpha}) \end{aligned}$$

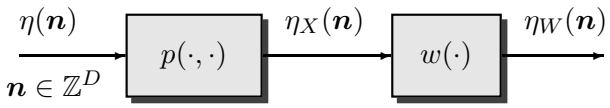


Figure 6.1: *Filtering of a random process with a space-varying filter followed by a space-invariant one.*

$$
\begin{aligned}
&= \sigma^2 \sum_{\boldsymbol{k}} \sum_{\boldsymbol{h}} e^{j2\pi(\boldsymbol{k}+\boldsymbol{h})^T \boldsymbol{V}^{-1}\boldsymbol{n}} e^{j2\pi \boldsymbol{h}^T \boldsymbol{V}^{-1}\boldsymbol{m}} \sum_{\boldsymbol{\alpha}} w(\boldsymbol{\alpha}) e^{-j2\pi \boldsymbol{k}^T \boldsymbol{V}^{-1}\boldsymbol{\alpha}} \\
&\quad \sum_{\boldsymbol{\beta}} w(\boldsymbol{\beta}) e^{-j2\pi \boldsymbol{h}^T \boldsymbol{V}^{-1}\boldsymbol{\beta}} p_{\boldsymbol{k}} * p_{\boldsymbol{h}}(\boldsymbol{m}-\boldsymbol{\beta}-\boldsymbol{\alpha}) \\
&= \sigma^2 \sum_{\boldsymbol{k}} \sum_{\boldsymbol{h}} e^{j2\pi(\boldsymbol{k}+\boldsymbol{h})^T \boldsymbol{V}^{-1}\boldsymbol{n}} e^{j2\pi \boldsymbol{h}^T \boldsymbol{V}^{-1}\boldsymbol{m}} w_{\boldsymbol{k}(-)} * w_{\boldsymbol{h}} * p_{\boldsymbol{k}} * p_{\boldsymbol{h}}(\boldsymbol{m}), \qquad (6.6)
\end{aligned}
$$

where $w_{\boldsymbol{h}}(\boldsymbol{m}) = w(\boldsymbol{m}) e^{-j2\pi \boldsymbol{h}^T \boldsymbol{V}^{-1}\boldsymbol{m}}$ and $w_{\boldsymbol{k}(-)}(\boldsymbol{m}) = w_{\boldsymbol{k}}(-\boldsymbol{m})$.

It follows that $\eta_W(\boldsymbol{n})$ is ciclostationary and $\sigma^2_{\eta_W}(\boldsymbol{n})$ can be calculated at each location $\boldsymbol{n}$ using (6.6) as

$$\sigma^2_{\eta_W}(\boldsymbol{n}) = r_{\eta_W}(\boldsymbol{n}, 0). \qquad (6.7)$$

Example. Suppose that $\boldsymbol{n} \in \mathbb{Z}^2$ and the process $\eta(\boldsymbol{n})$ is filtered with a LPSV filter that admits only two states. Let the periodicity be expressed by a quincunx lattice, that is with the periodicity matrix

$$\boldsymbol{V} = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}. \qquad (6.8)$$

The region $\mathcal{I}_{\boldsymbol{V}}$ has $|\det(\boldsymbol{V})| = 2$ vectors, namely $\boldsymbol{k}_0 = [0\ 0]^T$ and $\boldsymbol{k}_1 = [0\ 1]^T$. We denote $p_{\boldsymbol{k}_0}(\boldsymbol{n}) = p_0(\boldsymbol{n})$ and $p_{\boldsymbol{k}_1}(\boldsymbol{n}) = p_1(\boldsymbol{n})$. Therefore Equation (6.6) becomes

$$
\begin{aligned}
r_{\eta_W}(\boldsymbol{n}, \boldsymbol{m}) &= \sigma^2 \big(w_- * w * p_0 * p_0(\boldsymbol{m}) + e^{j\pi[1\ 1]\boldsymbol{n}} e^{j\pi[1\ 1]\boldsymbol{m}} w_- * w_I * p_1 * p_0(\boldsymbol{m}) \\
&\quad + e^{j\pi[1\ 1]\boldsymbol{n}} w_{I(-)} * w * p_1 * p_0(\boldsymbol{m}) + e^{j\pi[1\ 1]\boldsymbol{m}} w_{I(-)} * w_I * p_1 * p_1(\boldsymbol{m}) \big), \qquad (6.9)
\end{aligned}
$$

where $w_I(\boldsymbol{m}) = w(\boldsymbol{m}) e^{-j\pi[1\ 1]\boldsymbol{m}}$. The variance of $\eta_W(\boldsymbol{n})$ is given by

$$
\begin{aligned}
\sigma^2_{\eta_W}(\boldsymbol{n}) &= \sigma^2 \Big(w_- * w * p_0 * p_0(\boldsymbol{0}) + e^{j\pi[1\ 1]\boldsymbol{n}} \big(w_- * w_I + w_{I(-)} * w \big) * p_0 * p_1(\boldsymbol{0}) \\
&\quad + w_{I(-)} * w_I * p_1 * p_1(\boldsymbol{0}) \Big). \qquad (6.10)
\end{aligned}
$$

□

6.3 Denoising procedure

We assume the same observation model presented in (5.11) in the previous chapter. In particular, we consider that the sensor introduces an additive gaussian white noise $\eta(\boldsymbol{n})$. As in Section 5.2.3, we use a regularization approach to design the optimal filters for demosaicking.

Due to the data sampling structure of the CFA, these filters are periodical space-varying and the number of different states depends on the periodicity of the color filter array arrangement. Therefore, the noise in the color image after demosaicking is not white and its autocorrelation function is given by (6.5), where the $p_{\boldsymbol{k}}(\cdot)$ are the polyphase components of the filters.

Suppose to apply an undecimated wavelet transform to the noisy color image obtained by filtering the sensor data with the space-varying filters obtained by quadratic regularization as in Section 5.2.3. Denoting with $\hat{X}(\boldsymbol{n})$, for $X = R, G, B$, a reconstructed color component and with $\hat{X}_W(\boldsymbol{n})$ a detail wavelet subband of $\hat{X}(\boldsymbol{n})$, we have that $\hat{X}_W(\boldsymbol{n}) = w * \hat{X}(\boldsymbol{n})$, where $w(\cdot)$ is a space-invariant equivalent filter. Therefore at each subband the noise autocorrelation function is given by (6.6), so it is evident that the noise is ciclostationary and its variance can be calculated at each location $\boldsymbol{n}$ using (6.7).

This result can be used to devise a wavelet-based denoising procedure. In this chapter we consider a LMMSE (Linear Minimum Mean Squared Error) approach combined with a hard-thresholding. Each detail wavelet coefficient $\hat{X}_W(\boldsymbol{n})$ can be expressed as

$$\hat{X}_W(\boldsymbol{n}) = X_W(\boldsymbol{n}) + \eta_W(\boldsymbol{n}), \tag{6.11}$$

where $X_W(\boldsymbol{n})$ is the corresponding wavelet coefficient of the noise-free color component. Since $X_W(\boldsymbol{n})$ and $\eta_W(\boldsymbol{n})$ are mutually uncorrelated and both zero mean, the LMMSE estimate of $X_W(\boldsymbol{n})$ from $\hat{X}_W(\boldsymbol{n})$ is

$$\tilde{X}_W(\boldsymbol{n}) = \frac{\sigma^2_{X_W}(\boldsymbol{n})}{\sigma^2_{X_W}(\boldsymbol{n}) + \sigma^2_{\eta_W}(\boldsymbol{n})} \hat{X}_W(\boldsymbol{n}) \tag{6.12}$$

where $\sigma^2_{\eta_W}(\boldsymbol{n})$ is obtained using (6.7), while the variance of $X_W(\boldsymbol{n})$ is estimated as $\sigma^2_{X_W}(\boldsymbol{n}) = \sigma^2_{\hat{X}_W}(\boldsymbol{n}) - \sigma^2_{\eta_W}(\boldsymbol{n})$. The variance $\sigma^2_{\hat{X}_W}(\boldsymbol{n})$ is computed from the wavelet coefficients in a 5×5 window $\mathcal{B}(\boldsymbol{n})$ centered at $\boldsymbol{n}$,

$$\sigma^2_{\hat{X}_W}(\boldsymbol{n}) = \sum_{\boldsymbol{m} \in \mathcal{B}(\boldsymbol{n})} \hat{X}^2_W(\boldsymbol{m}). \tag{6.13}$$

Successively the wavelet coefficients are thresholded with the hard-threshold rule (see Chapter 5),

$$\tilde{X}_W(\boldsymbol{n}) = \text{sgn}\{\hat{X}_W(\boldsymbol{n})\} \max\{|\hat{X}_W(\boldsymbol{n}) - T(\boldsymbol{n})|, 0\}, \tag{6.14}$$

with $T(\boldsymbol{n}) = \text{const} \cdot \sigma_{\eta_W}(\boldsymbol{n})$, and inverse wavelet transform is performed to obtain the denoised component $\tilde{X}(\boldsymbol{n})$.

After denoising, the adaptive method described in Section 5.2.4 can be applied. In this way the edge-estimation is performed on the denoised image, so it is not affected by the noise.

In detail, the proposed algorithm can be summarized as follows:

- estimation of the demosaicked color components $\hat{R}, \hat{G}, \hat{B}$ using the filters designed with the quadratic regularization described in Section 5.2.3;

- undecimated wavelet transform of $\hat{R}, \hat{G}, \hat{B}$;

- denoising of the coefficients of the color components with the LMMSE approach combined with hard-thresholding as described above;

- inverse undecimated wavelet transform of the denoised components;

- adaptive reconstruction of the color components according to the scheme presented in Section 5.2.4, performing the edge-estimation on the denoised color components.

6.4 Denoising of the luminance component

For the pure-color CFA arrangements a more efficient strategy is proposed. In fact, the technique described in Section 6.3 requires to wavelet transform and denoise three color component, that is computationally demanding. Exploiting the luminance-based reconstruction approaches proposed in Chapter 5, it is possible to overcome this problem.

We start estimating a noisy version of the luminance component $L(\boldsymbol{n})$ from the CFA data with the space-varying filter G_ℓ obtained in Section 5.2.3. Then, we apply an undecimated wavelet transform and each wavelet coefficient can be expressed as

$$\hat{L}_W(\boldsymbol{n}) = L_W(\boldsymbol{n}) + \eta_W(\boldsymbol{n}). \tag{6.15}$$

We denoise them with the LMMSE approach combined with hard-thresholding described above for the color components. The noise variance at each subband is estimated with the same approach presented in Section 6.3. Note that, if the Bayer pattern is used as CFA arrangement, the autocorrelation function of the noise is computed as in the example given in Section 6.2, and the variance can be expressed as in (6.10). Successively, the denoised luminance component $\tilde{L}(\boldsymbol{n})$ is reconstructed with an inverse undecimated wavelet transform. The noise $\eta(\boldsymbol{n})$ is estimated by $\tilde{\eta}(\boldsymbol{n}) = \hat{L}(\boldsymbol{n}) - \tilde{L}(\boldsymbol{n})$ and this estimate is used to denoise the CFA image, that is

$$\tilde{\mathcal{I}}_s(\boldsymbol{n}) = \mathcal{I}_s(\boldsymbol{n}) - \tilde{\eta}(\boldsymbol{n}) \tag{6.16}$$

as reported in [144]. Then, the quadratic estimation of the luminance is improved with the adaptive method developed in Section 5.2.4. Finally, the color components are reconstructed using bilinear interpolation of the values $\tilde{L}(\boldsymbol{n}) - \tilde{X}(\boldsymbol{n})$, where $\tilde{X}(\boldsymbol{n})$, for $X = R, G, B$, are the available color samples taken from the denoised CFA image.

In detail, the proposed luminance-based technique can be summarized as follows:

- estimation of the luminance component $\hat{L}(\boldsymbol{n})$ using the filter $G_\ell(\boldsymbol{n})$ designed with the quadratic regularization described in Section 5.2.3;

- undecimated wavelet transform of $\hat{L}(\boldsymbol{n})$;

- denoising of the coefficients of the luminance component with the LMMSE approach combined with hard-thresholding described in Section 6.3;

- inverse undecimated wavelet transform of the denoised luminance component;

Reconstruction Methods	$\sigma = 0$	$\sigma = 5$	$\sigma = 10$	$\sigma = 15$
Demosaicking [87] + Denoising [119]	39.27	38.52	31.38	29.33
Denoising [142] + Demosaicking [87]	39.27	34.76	31.90	29.96
Joint Demosaicking and Denoising [65]	32.28	31.35	29.67	28.30
Joint Demosaicking and Denoising [109]	36.05	34.52	31.73	29.84
Joint Demosaicking and Denoising [144]	39.40	35.28	32.00	29.92
Proposed Approach	39.57	35.15	31.84	29.65

Table 6.1: *Average CPSNR (dB) for 20 test-images with different demosaicking-denoising solutions.*

- estimation of the noise as $\tilde{\eta}(\boldsymbol{n}) = \hat{L}(\boldsymbol{n}) - \tilde{L}(\boldsymbol{n})$ and denoising of the CFA image using (6.16);

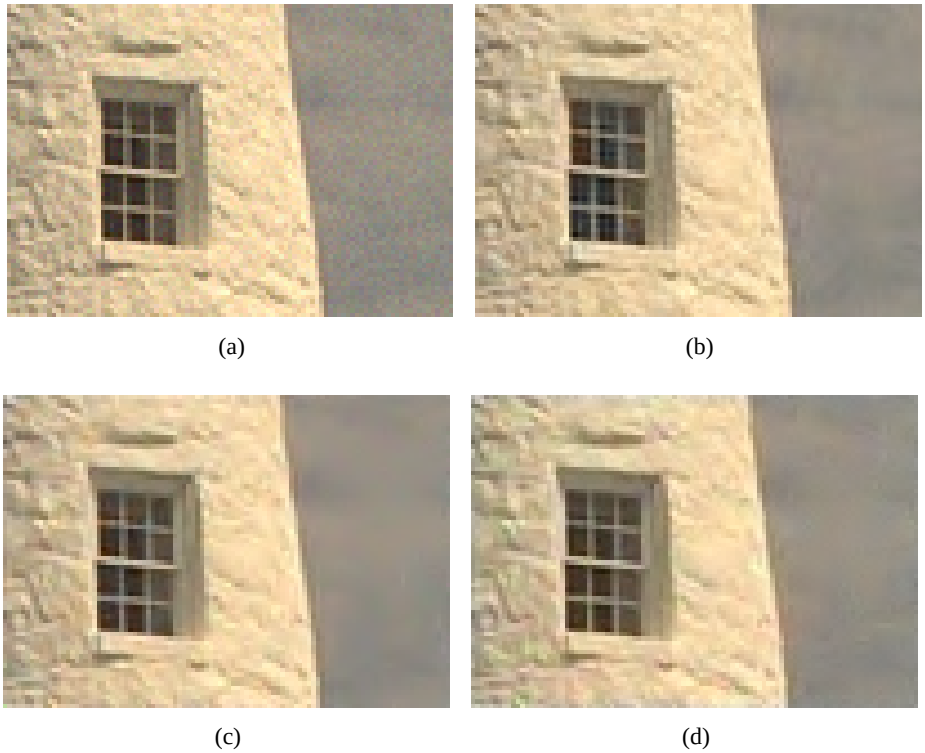
- adaptive reconstruction of the luminance component according to the scheme presented in Section 5.2.4, performing the edge-estimation on the denoised luminance component;

- reconstruction of the color components through bilinear interpolation of $\tilde{L}(\boldsymbol{n}) - \tilde{X}(\boldsymbol{n})$, for $X = R, G, B$, exploiting the denoised CFA image.

6.5 Experimental results

Table 6.1 reports the CPSNR values obtained with different solutions for demosaicking and denoising, using the MATLAB code provided by the authors. CPSNR values are averaged over 20 test images of the Kodak dataset [53], corrupted with different standard deviation noise levels $\sigma = 0, 5, 10$ and 15, and sampled using the Bayer pattern. The demosaicking algorithm of [87] followed by the denoising presented in [119], the CFA-denoising of [142] followed by the demosaicking procedure [87], the joint demosaicking and denoising techniques proposed in [65, 144], are compared with the proposed technique, in which demosaicking is performed using the luminance-based adaptive method, while denoising is done using the LMMSE approach described in the previous section. The Daubechies' wavelet with two vanishing moments is employed, with three levels of decomposition. Fig. 6.2(a) shows a portion of the original test image *lighthouse* corrupted by additive noise with $\sigma = 5$, Fig. 6.2(b) the resulting image after reconstruction and denoising with the method of [144], Fig. 6.2(c)-(d) using the proposed approach, simulating the Bayer pattern sampling and one of the CFA arrangements described in [66], respectively.

We can see that the proposed method outperforms demosaicking followed by denoising, and its performances are superior or comparable with the results obtained by the other joint demosaicking and denoising techniques, which have a higher computational cost. Moreover, the proposed approach can be applied to any CFA arrangement.



(a) (b)

(c) (d)

Figure 6.2: *Portion of the image no. 16 of the Kodak set: (a) Noisy image, $\sigma = 5$; (b) image reconstructed and denoised by approach [144]; (c) image reconstructed and denoised by the proposed approach; (d) image reconstructed and denoised by the proposed approach, after the CFA-sampling proposed in [66].*

Chapter 7

Optimal image interpolation based on local statistics

In this chapter, we propose a new adaptive approach for image interpolation that is able to increase the resolution without distorting or smoothing the edges. An optimal estimation of the high-resolution image is performed using a local statistical model, in order to adapt the reconstruction to the features of the image. An improvement of the interpolation using the bilateral filters is also described. The experimental results show that the proposed approach is effective and, in most cases, outperforms other adaptive techniques both in terms of PSNR and in subjective quality.

7.1 Introduction

In the last years, we observed a large diffusion of digital images in the everyday life, and electronic devices able to capture digital images, such as digital cameras, PDA, cell phones, etc., have become very popular. Often, however, the quality of the images is not as good as desired. In particular, the *resolution* of the image has to be limited due to the cost of the acquisition device and to reduce the amount of data that has to be stored and/or transmitted. Therefore, in many situations it is not possible to detect or recognize small details. This is a problem both for the human observers and for the computer vision algorithms applied to the images.

The procedure to enlarge the size of the image is known as *zooming* or *image magnification*. It requires an interpolation of the available data of the lower resolution image in order to estimate the missing pixels and obtain a higher resolution image. The most common methods, such as the nearest neighbor replication or the bilinear and bicubic interpolation [78], often produce images with various artifacts along object boundaries, including moiré effects, blurring, and zigzagging edges. Motivated by these drawbacks, various adaptive interpolation algorithms have been proposed that adapt the estimation to the local behavior of the image and reduce the artifacts near the edges. In [5] a map of the high-resolution details is generated and is used to iteratively correct the interpolated pixels. In [21] a nonlinear approach is proposed based on the assumption that the edges of the low resolution image derive from steep edges of the high resolution version. In [8] Aly and Dubois present an image magnification method

incorporating the total-variation regularization, while in [28] the Lipschitz regularity of the wavelets is used to estimate the high resolution edges, and Mueller *et al.* exploit the properties of the contourlet transform to improve the regularity of the object boundaries in the estimated image [101]. Zhang and Wu in [141] propose to perform two directional interpolations, then a weighted sum of them is chosen. A Bayesian approach is considered by Schultz and Stevenson [128], while Li and Orchard [85] propose to estimate the covariance of the high-resolution image from its low-resolution version and to estimate the missing pixels with a Wiener filter. Despite of its simplicity, this approach proves to be able to estimate also the detailed regions of the image. Recently, a similar strategy is analyzed also in [147], where images are modeled as a piecewise autoregressive process and pixels are estimated with a soft-decision interpolation, and in [82], where a Markov random field model is considered. A kernel regression approach for use in image interpolation is proposed in [130].

In this chapter we extend the strategy described in [85] giving a more general formulation of the optimal estimation of the high-resolution image from a subsampled version. A local statistical model is used, in order to adapt the reconstruction to the local features of the image and improve the estimation of the details. The parameters of this statistical model are estimated from the available pixels of the low-resolution image. In particular, the presented approach can be applied for each integer expansion ratio, differently from many other techniques that permit only to increase the resolution by a factor 2 (or, iteratively, 2^n, where n is an integer). Moreover, it allows for different trade-off between quality of the reconstructed image and complexity of the algorithm, since the number of coefficients of the optimal filter can be chosen according to the available computational resources. The presented technique is also robust to the effects of noise.

The chapter is organized as follows. Section 7.2 reviews the procedure to obtain the optimal interpolation filter, while the local statistical model with a novel strategy to estimate its parameters from the low-resolution image is presented in Section 7.3. In Section 7.4 the details of the proposed algorithm are described and, since the human visual system is highly sensitive to the edges, in Section 7.4.1 the use of bilateral filters to improve the reconstruction near the boundaries is discussed. Results and comparisons with other interpolation methods are presented in Section 7.5.

7.2 Derivation of the optimal filter

Let $y(\boldsymbol{m})$ be a low-resolution (LR) image, with $\boldsymbol{m} \in \boldsymbol{\Gamma}$, where $\boldsymbol{\Gamma} \in \mathbb{R}^2$ is a lattice, from which a high-resolution (HR) image $x(\boldsymbol{n})$, with $n \in \boldsymbol{\Lambda}$, has to be estimated (see Fig. 7.1). The LR image can be considered as a downsampled version of the HR image, as assumed in many other image interpolation approaches [141, 85, 147, 82]. Then, $\boldsymbol{\Gamma}$ is a sublattice of $\boldsymbol{\Lambda}$ [38], $\boldsymbol{\Gamma} \subset \boldsymbol{\Lambda}$, and the relation between $y(\boldsymbol{m})$ and $x(\boldsymbol{n})$ is given by

$$y(\boldsymbol{m}) = x(\boldsymbol{m}) + \eta(\boldsymbol{m}), \qquad \boldsymbol{m} \in \boldsymbol{\Gamma} \tag{7.1}$$

where an independent and identically distributed (i.i.d.) zero mean noise $\eta(\boldsymbol{m})$ is also considered. Let $g(\boldsymbol{n}, \boldsymbol{\ell})$, $\boldsymbol{\ell} \in \mathcal{S}$, where $\mathcal{S}$ indicates the spatial support, a space-varying filter that

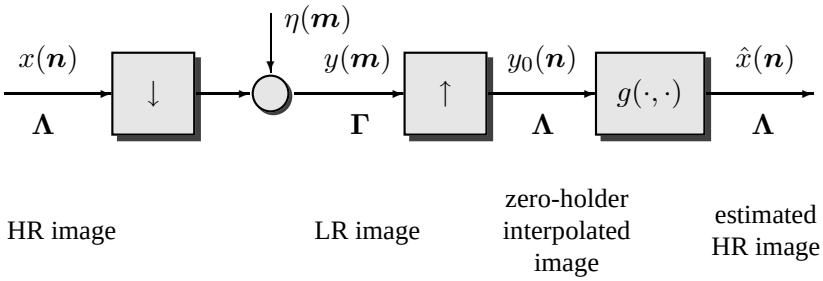


Figure 7.1: *Scheme of image sampling and interpolation.*

allows a linear estimate of the HR image from the available LR samples,

$$\hat{x}(\boldsymbol{n}) = \sum_{\boldsymbol{\ell}\in\mathcal{S}} g(\boldsymbol{n},\boldsymbol{\ell}) y_0(\boldsymbol{n}-\boldsymbol{\ell}), \tag{7.2}$$

with $y_0(\boldsymbol{n})$, $\boldsymbol{n}\in\boldsymbol{\Lambda}$, denoting a zero-holder interpolation of $y(\boldsymbol{m})$. Then, the optimal filter is obtained minimizing the average mean square error

$$E\left[|x(\boldsymbol{n})-\hat{x}(\boldsymbol{n})|^2\right], \tag{7.3}$$

and the minimization of (7.3) is found when the *orthogonality principle* holds [57],

$$E\left[(x(\boldsymbol{n})-\hat{x}(\boldsymbol{n}))y_0(\boldsymbol{n}-\boldsymbol{k})\right] = 0 \quad \forall \boldsymbol{k}\in\mathcal{S}. \tag{7.4}$$

Substituting (7.2) into the last equation we have

$$E\left[x(\boldsymbol{n})y_0(\boldsymbol{n}-\boldsymbol{k})\right] - \sum_{\boldsymbol{\ell}\in\mathcal{S}} g(\boldsymbol{n},\boldsymbol{\ell}) E\left[y_0(\boldsymbol{n}-\boldsymbol{\ell})y_0(\boldsymbol{n}-\boldsymbol{k})\right] = 0 \quad \forall \boldsymbol{k}\in\mathcal{S}, \tag{7.5}$$

and, using the relation (7.1),

$$\begin{aligned} &E\left[x(\boldsymbol{n})\big(x(\boldsymbol{n}-\boldsymbol{k})+\eta(\boldsymbol{n}-\boldsymbol{k})\big)\right] - \\ &\quad \sum_{\boldsymbol{\ell}\in\mathcal{S}\cap(\boldsymbol{\Gamma}+\boldsymbol{n})} g(\boldsymbol{n},\boldsymbol{\ell}) E\left[\big(x(\boldsymbol{n}-\boldsymbol{\ell})+\eta(\boldsymbol{n}-\boldsymbol{\ell})\big)\big(x(\boldsymbol{n}-\boldsymbol{k})+\eta(\boldsymbol{n}-\boldsymbol{k})\big)\right] = 0, \end{aligned} \tag{7.6}$$

$\forall \boldsymbol{k}\in\mathcal{S}\cap(\boldsymbol{\Gamma}+\boldsymbol{n})$, where $\boldsymbol{\Gamma}+\boldsymbol{n} = \{\boldsymbol{c}+\boldsymbol{n} \,|\, \boldsymbol{c}\in\boldsymbol{\Gamma}\}$ is a coset of $\boldsymbol{\Gamma}$ in $\boldsymbol{\Lambda}$ [38]. Assuming that $x(\boldsymbol{n})$ is a locally stationary random process, i.e. its autocorrelation $r_x(\boldsymbol{n},\boldsymbol{k}) = E\left[x(\boldsymbol{n})x(\boldsymbol{n}-\boldsymbol{k})\right]$ is constant over a large enough neighborhood of the pixel $\boldsymbol{n}$ such that it contains the support $\mathcal{S}$, and denoting with $r_\eta(\boldsymbol{k})$ the autocorrelation of the stationary noise η, equation (7.6) becomes

$$r_x(\boldsymbol{n},\boldsymbol{k}) - \sum_{\boldsymbol{\ell}\in\mathcal{S}\cap(\boldsymbol{\Gamma}+\boldsymbol{n})} g(\boldsymbol{n},\boldsymbol{\ell})\big(r_x(\boldsymbol{n},\boldsymbol{k}-\boldsymbol{\ell}) + r_\eta(\boldsymbol{k}-\boldsymbol{\ell})\big) = 0 \quad \forall \boldsymbol{k}\in\mathcal{S}\cap(\boldsymbol{\Gamma}+\boldsymbol{n}). \tag{7.7}$$

Denoting with $d_{\boldsymbol{n}}$ the cardinality of $\mathcal{S}\cap(\boldsymbol{\Gamma}+\boldsymbol{n})$, (7.7) corresponds to a set of $d_{\boldsymbol{n}}$ linear equations in the $d_{\boldsymbol{n}}$ unknown values $g(\boldsymbol{n},\boldsymbol{\ell})$, $\boldsymbol{\ell}\in\mathcal{S}\cap(\boldsymbol{\Gamma}+\boldsymbol{n})$, for each pixel $\boldsymbol{n}$. In matrix form we obtain

$$\left(\boldsymbol{R}_{x(\boldsymbol{n})} + \boldsymbol{R}_\eta\right)\, \boldsymbol{g}_{\boldsymbol{n}} = \boldsymbol{r}_{x(\boldsymbol{n})}, \tag{7.8}$$

where

- $\boldsymbol{g_n}$ is a column vector formed by the $d_{\boldsymbol{n}}$ unknowns $g(\boldsymbol{n}, \boldsymbol{\ell})$, for $\boldsymbol{\ell} \in \mathcal{S} \cap (\boldsymbol{\Gamma} + \boldsymbol{n})$, (e.g. using a columnwise ordering), that is $\boldsymbol{g_n} = [g(\boldsymbol{n}, \boldsymbol{\ell}_1), g(\boldsymbol{n}, \boldsymbol{\ell}_2), \cdots, g(\boldsymbol{n}, \boldsymbol{\ell}_{d_{\boldsymbol{n}}})]^T$;

- $\boldsymbol{r}_{x(\boldsymbol{n})}$ is a column vector given by the $d_{\boldsymbol{n}}$ values $r_x(\boldsymbol{n}, \boldsymbol{k})$, for $\boldsymbol{k} \in \mathcal{S} \cap (\boldsymbol{\Gamma} + \boldsymbol{n})$, (with the same element ordering used for $\boldsymbol{g_n}$), $\boldsymbol{r}_{x(\boldsymbol{n})} = [r_x(\boldsymbol{n}, \boldsymbol{k}_1), r_x(\boldsymbol{n}, \boldsymbol{k}_2), \cdots, r_x(\boldsymbol{n}, \boldsymbol{k}_{d_{\boldsymbol{n}}})]^T$;

- $\boldsymbol{R}_{x(\boldsymbol{n})}$ is a $d_{\boldsymbol{n}} \times d_{\boldsymbol{n}}$ matrix where the (i, j)-element is $r_x(\boldsymbol{n}, \boldsymbol{k}_i - \boldsymbol{\ell}_j)$. If a columnwise ordering is chosen for $\boldsymbol{g_n}$ and $\boldsymbol{r}_{x(\boldsymbol{n})}$, the matrix $\boldsymbol{R}_{x(\boldsymbol{n})}$ is a Toeplitz-block Toeplitz matrix;

- $\boldsymbol{R}_\eta$ is the $d_{\boldsymbol{n}} \times d_{\boldsymbol{n}}$ autocorrelation matrix of the noise. Since the noise is assumed i.i.d., $\boldsymbol{R}_\eta$ is diagonal.

Note that the number of the unknowns and the size of the linear system depend on the position $\boldsymbol{n}$, the chosen filter support $\mathcal{S}$, and the lattice $\boldsymbol{\Gamma}$.

Equation (7.8) allows to compute the coefficients of the optimal filter that minimizes the mean square error in (7.3). However, this linear estimation requires the knowledge of an accurate statistical model for the image $x(\boldsymbol{n})$. This is an open problem because natural images are not stationary and their characteristics strongly depend on the local behavior of the edges. In this chapter we propose to apply a local autocorrelation model that accounts for the characteristics of the image in the neighborhood of each missing pixel. In the next section we describe this model and the procedure to estimate its parameters from a sampled version of the original image.

7.3 Local Image Statistical Model

Usually images are modeled as stationary and zero-mean random fields. A common model to describe them is the first-order Markov process for which the correlation between the pixels is proportional to their geometric separation only [71]. The autocorrelation function for the two-dimensional Markov model is

$$r_x(\boldsymbol{k}) = \sigma_x^2 \exp\left(-\omega_0 \sqrt{\boldsymbol{k}^T \boldsymbol{k}}\right), \tag{7.9}$$

for given variance σ_x^2 and parameter $\omega_0 \geq 0$. However, in natural images the local correlation is not isotropic but depends on the orientation of the edges. Therefore, it is preferable to consider an *anisotropic* model, such as

$$r_x(\boldsymbol{n}, \boldsymbol{k}) = \sigma_x^2(\boldsymbol{n}) \exp\left(-\sqrt{\boldsymbol{k}^T \boldsymbol{A}^T(\boldsymbol{n}) \boldsymbol{K}(\boldsymbol{n}) \boldsymbol{A}(\boldsymbol{n}) \boldsymbol{k}}\right), \tag{7.10}$$

where

$$\boldsymbol{K}(\boldsymbol{n}) = \begin{bmatrix} \omega_1^2(\boldsymbol{n}) & 0 \\ 0 & \omega_2^2(\boldsymbol{n}) \end{bmatrix}, \tag{7.11}$$

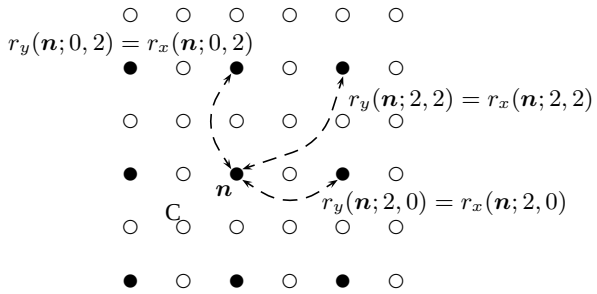


Figure 7.2: *Spatial configuration of the pixel of the LR image (black dots) and of the missing pixels of the HR image (white dots) when* $z = 2$.

with $\omega_1^2(\boldsymbol{n}) \geq 0$, $\omega_2^2(\boldsymbol{n}) \geq 0$, and

$$\boldsymbol{A}(\boldsymbol{n}) = \begin{bmatrix} \cos\theta(\boldsymbol{n}) & \sin\theta(\boldsymbol{n}) \\ -\sin\theta(\boldsymbol{n}) & \cos\theta(\boldsymbol{n}) \end{bmatrix}. \tag{7.12}$$

In this way the correlation is not constant over all the directions but depends on the orientation represented by $\theta(\boldsymbol{n})$. The autocorrelation with a one-position shift is $\sigma^2(\boldsymbol{n})\exp(-\omega_1(\boldsymbol{n}))$ along the direction given by the angle $\theta(\boldsymbol{n})$, while is $\sigma^2(\boldsymbol{n})\exp(-\omega_2(\boldsymbol{n}))$ in the orthogonal direction. Countours of equal correlation form ellipses where the angle $\theta(\boldsymbol{n})$ represents the orientation of one of the two principal axes. The two axes are proportional to $\omega_1^{-1}(\boldsymbol{n})$ and $\omega_2^{-1}(\boldsymbol{n})$. The same local statistical models have been used in [20, 24] for the purpose of image compression, and in [130, 113] into the framework of kernel regression and normalized convolution, respectively.

7.3.1 Estimation of the parameters from the low-resolution images

The described anisotropic model requires four parameters to be completely specified in each pixel $\boldsymbol{n}$, namely $\sigma_x^2(\boldsymbol{n})$, $\theta(\boldsymbol{n})$, $\omega_1(\boldsymbol{n})$, and $\omega_2(\boldsymbol{n})$. These parameters can be estimated by the low-resolution image with the following approach.

Suppose that an increase of the resolution by an integer factor z both horizontally and vertically is required. Then, the high-resolution lattice $\boldsymbol{\Lambda}$ can be set as $\boldsymbol{\Lambda} = \mathbb{Z}^2$ and the lattice $\boldsymbol{\Gamma}$ as generated by the sampling matrix $\boldsymbol{V}$,

$$\boldsymbol{V} = \begin{bmatrix} z & 0 \\ 0 & z \end{bmatrix}. \tag{7.13}$$

The set of the available samples of the LR image $y(\boldsymbol{m})$ forms a subset of the pixels of the HR image $x(\boldsymbol{n})$ divided by z locations. For instance, Fig. 7.2 shows the spatial configuration of the pixels of the LR image (represented with the black dots) with respect to the positions of the HR image that has to be estimated (white dots) when $z = 2$.

The correlation function of the LR image $y(\boldsymbol{m})$ is related to the correlation of $x(\boldsymbol{n})$ as

$$\begin{aligned} r_y(\boldsymbol{m},\boldsymbol{k}) &= E\left[y(\boldsymbol{m})y^*(\boldsymbol{m}+\boldsymbol{k})\right] \\ &= E\left[\left(x(\boldsymbol{m})+\eta(\boldsymbol{m})\right)\left(x^*(\boldsymbol{m}+\boldsymbol{k})+\eta^*(\boldsymbol{m}+\boldsymbol{k})\right)\right] \\ &= E\left[x(\boldsymbol{m})x^*(\boldsymbol{m}+\boldsymbol{k})\right]+E\left[\eta(\boldsymbol{m})\eta^*(\boldsymbol{m}+\boldsymbol{k})\right] \\ &= r_x(\boldsymbol{m},\boldsymbol{k})+r_\eta(\boldsymbol{k}), \end{aligned} \tag{7.14}$$

with $\boldsymbol{m},\boldsymbol{k}\in\boldsymbol{\Gamma}$. Assuming that x is a locally stationary process, as done in Section 7.2, its autocorrelation function $r_x(\boldsymbol{n},\boldsymbol{k})$, with $\boldsymbol{n}\in\boldsymbol{\Lambda},\boldsymbol{k}\in\boldsymbol{\Gamma}$, can be computed by averaging the available values $r_x(\boldsymbol{m},\boldsymbol{k})$, for $\boldsymbol{m}\in\boldsymbol{\Gamma}$, that is

$$r_x(\boldsymbol{n},\boldsymbol{k})=\sum_{\boldsymbol{i}\in\mathcal{S}_a\cap(\boldsymbol{\Gamma}+\boldsymbol{n})} a(\boldsymbol{i})r_x(\boldsymbol{n}-\boldsymbol{i},\boldsymbol{k}) \quad \boldsymbol{n}\in\boldsymbol{\Lambda},\boldsymbol{k}\in\boldsymbol{\Gamma}, \tag{7.15}$$

where $a(\cdot)$ is an averaging kernel with a limited support $\mathcal{S}_a$ and $\sum_{\boldsymbol{i}\in\mathcal{S}_a\cap(\boldsymbol{\Gamma}+\boldsymbol{n})} a(\boldsymbol{i})=1$.

Equation (7.15) gives

$$\begin{aligned} r_x(\boldsymbol{n},\boldsymbol{k}) &= \sum_{\boldsymbol{i}\in\mathcal{S}_a\cap(\boldsymbol{\Gamma}+\boldsymbol{n})} a(\boldsymbol{i})r_y(\boldsymbol{n}-\boldsymbol{i},\boldsymbol{k}) - r_\eta(\boldsymbol{k})\sum_{\boldsymbol{i}\in\mathcal{S}_a\cap(\boldsymbol{\Gamma}+\boldsymbol{n})} a(\boldsymbol{i}) \\ &= r_y^a(\boldsymbol{n},\boldsymbol{k})-r_\eta(\boldsymbol{k}), \end{aligned} \tag{7.16}$$

where $r_y^a(\boldsymbol{n},\boldsymbol{k})=\sum a(\boldsymbol{i})r_y(\boldsymbol{n}-\boldsymbol{i},\boldsymbol{k})$. In this chapter, we use as averaging kernel the bilinear operator, that is, if $z=2$,

$$a=\begin{bmatrix} 0.25 & 0.5 & 0.25 \\ 0.5 & 1 & 0.5 \\ 0.25 & 0.5 & 0.25 \end{bmatrix}. \tag{7.17}$$

Relation (7.16) allows to estimate the correlation values between the pixels $\boldsymbol{n}$ of the lattice $\boldsymbol{\Lambda}$ that belong to the same coset $\boldsymbol{n}+\boldsymbol{\Gamma}$. From these values it is possible to compute the parameters that drive the correlation model (7.10), fully describing the autocorrelation of the image $x(\boldsymbol{n})$. In fact, reminding that for the stationary noise $r_\eta(\boldsymbol{k})=\sigma_\eta^2\delta(\boldsymbol{k})$, we obtain

$$\sigma_x^2(\boldsymbol{n})=r_y^a(\boldsymbol{n},\boldsymbol{0})-\sigma_\eta^2, \tag{7.18}$$

and

$$r_y^a(\boldsymbol{n};z,0) = r_x(\boldsymbol{n};z,0)=\sigma_x^2(\boldsymbol{n})e^{-z\sqrt{\omega_1^2(\boldsymbol{n})\cos^2\theta(\boldsymbol{n})+\omega_2^2(\boldsymbol{n})\sin^2\theta(\boldsymbol{n})}}, \tag{7.19}$$

$$r_y^a(\boldsymbol{n};0,z) = r_x(\boldsymbol{n};0,z)=\sigma_x^2(\boldsymbol{n})e^{-z\sqrt{\omega_1^2(\boldsymbol{n})\sin^2\theta(\boldsymbol{n})+\omega_2^2(\boldsymbol{n})\cos^2\theta(\boldsymbol{n})}}, \tag{7.20}$$

$$r_y^a(\boldsymbol{n};z,z) = r_x(\boldsymbol{n};z,z)=\sigma_x^2(\boldsymbol{n})e^{-z\sqrt{\omega_1^2(\boldsymbol{n})+\omega_2^2(\boldsymbol{n})+2(\omega_2^2(\boldsymbol{n})-\omega_1^2(\boldsymbol{n}))\sin\theta(\boldsymbol{n})\cos\theta(\boldsymbol{n})}}, \tag{7.21}$$

where, with a slight change of notation, $r_x\left(\boldsymbol{n},\begin{bmatrix}k_1\\k_2\end{bmatrix}\right)=r_x(\boldsymbol{n};k_1,k_2)$.

With simple mathematical operations we find

$$\theta(\boldsymbol{n}) = \frac{1}{2}\text{atan}\left(\frac{c_3 - c_1 - c_2}{c_2 - c_1}\right) \pm \frac{\pi}{2} \tag{7.22}$$

$$\omega_2(\boldsymbol{n}) = \sqrt{\frac{1}{2}\left(c_1 + c_2 - \frac{c_1 - c_2}{\cos 2\theta(\boldsymbol{n})}\right)} \tag{7.23}$$

$$\omega_1(\boldsymbol{n}) = \sqrt{c_1 + c_2 - \omega_2^2(\boldsymbol{n})}, \tag{7.24}$$

where c_1, c_2, and c_3 are $c_1 = (\log(r_y^a(\boldsymbol{n}; z, 0)/\sigma_x^2(\boldsymbol{n}))/z)^2$, $c_2 = (\log(r_y^a(\boldsymbol{n}; 0, z)/\sigma_x^2(\boldsymbol{n}))/z)^2$, and $c_3 = (\log(r_y^a(\boldsymbol{n}; z, z)/\sigma_x^2(\boldsymbol{n}))/z)^2$, respectively.

The two solutions for the angle $\theta(\boldsymbol{n})$ given by (7.22) correspond to a swap of the two values of $\omega_1(\boldsymbol{n})$ and $\omega_2(\boldsymbol{n})$; therefore, they give the same autocorrelation function.

7.4 Implementation Details

As described in Section 7.3, it is possible to estimate the parameters of the local statistical model presented in (7.10) for the HR image directly from the known LR values. This allows to compute the optimal filter for the reconstruction of each pixel of the HR image with an adaptive strategy, where the filter coefficients depend on the correlations between the pixels estimated from the LR image.

Then, the proposed algorithm can be summarized as follows:

- for each pixel $\boldsymbol{m}$ of the LR image
  - estimate the variance and the horizontal, vertical, and diagonal correlation of $y(\boldsymbol{m})$, namely $r_y(\boldsymbol{m}; 0, 0)$, $r_y(\boldsymbol{m}; 0, z)$, $r_y(\boldsymbol{m}; z, 0)$, and $r_y(\boldsymbol{m}; z, z)$;

  end.

- for each pixel $\boldsymbol{n}$ of the HR image
  - compute $r_x(\boldsymbol{n}; 0, 0)$, $r_x(\boldsymbol{n}; 0, z)$, $r_x(\boldsymbol{n}; z, 0)$, and $r_x(\boldsymbol{n}; z, z)$ using (7.16);
  - estimate $\theta(\boldsymbol{n})$, $\omega_1(\boldsymbol{n})$, and $\omega_2(\boldsymbol{n})$ using (7.22)-(7.24);
  - compute the autocorrelation matrices $\boldsymbol{R}_{x(\boldsymbol{n})}$ and $\boldsymbol{r}_{x(\boldsymbol{n})}$ according to the anisotropic model in (7.10);
  - compute the coefficients of the optimal filter $g(\boldsymbol{n}, \boldsymbol{\ell})$ as in (7.8);
  - estimate $\hat{x}(\boldsymbol{n}) = \sum_{\boldsymbol{\ell}} g(\boldsymbol{n}, \boldsymbol{\ell}) y(\boldsymbol{n} - \boldsymbol{\ell})$.

  end.

The main drawback of this approach is the complexity of the inversion of the $d_{\boldsymbol{n}} \times d_{\boldsymbol{n}}$ matrix $(\boldsymbol{R}_{x(\boldsymbol{n})} + \boldsymbol{R}_{\eta})$ for each pixel, in order to compute the filter coefficients. However, since the matrix $\boldsymbol{R}_{x(\boldsymbol{n})}$ has a Toeplitz-block Toeplitz structure, the inversion can be computed in a fast way with the Levinson-Durbin procedure [57]. Moreover, it has to be observed that, due to the

symmetry of the statistical model described in (7.10), the reconstruction filter is symmetric, that is $g(\boldsymbol{n}, \boldsymbol{\ell}) = g(\boldsymbol{n}, -\boldsymbol{\ell})$, therefore the knowledge of only $d_{\boldsymbol{n}}/2$ coefficients is necessary. Hence, only a $d_{\boldsymbol{n}}/2 \times d_{\boldsymbol{n}}/2$ matrix has to be inverted.

For instance, let us consider an increase of resolution by a factor $z = 2$. We analyze the estimation of the image values in the positions $\boldsymbol{c}$ where the four known neighbors are located along the diagonals (as the pixel labeled with C in Fig. 7.2), using a filter with a 3×3 spatial support, that is with four coefficients $g(\boldsymbol{c}; \pm 1, \pm 1)$. Note that the same situation is considered in many other image interpolation approaches, such as [141, 85, 147]. Exploiting the symmetry of the statistical model, the filter coefficients are obtained as

$$\begin{bmatrix} g(\boldsymbol{c}; -1, -1) \\ g(\boldsymbol{c}; -1, +1) \end{bmatrix} = \begin{bmatrix} g(\boldsymbol{c}; +1, +1) \\ g(\boldsymbol{c}; +1, -1) \end{bmatrix} =$$
$$= \begin{bmatrix} r_x(\boldsymbol{c}; 0, 0) + \sigma_\eta^2 + r_x(\boldsymbol{c}; -2, 2) & r_x(\boldsymbol{c}; 0, 2) + r_x(\boldsymbol{c}; 2, 0) \\ r_x(\boldsymbol{c}; 0, 2) + r_x(\boldsymbol{n}; 2, 0) & r_x(\boldsymbol{c}; 0, 0) + \sigma_\eta^2 + r_x(\boldsymbol{c}; 2, 2) \end{bmatrix}^{-1} \begin{bmatrix} r_x(\boldsymbol{c}; 1, 1) \\ r_x(\boldsymbol{c}; 1, -1) \end{bmatrix}.$$

7.4.1 Improvement using bilateral filters

The statistical model presented in Section 7.3 describes images that are locally stationary in the neighborhood of each pixel. Even though this assumption is useful for the design of the optimal filter, it is not verified for natural images. In particular, the edges can be seen as non-stationary elements and can affect the performance of a statistical approach. In fact, especially near sharp edges, the reconstruction filter takes into account also the values of pixels whose statistical characteristics are considerably different from the characteristics of the pixel $\hat{x}(\boldsymbol{n})$ that has to be estimated.

A solution to improve the quality of the interpolated image is to assign bigger relevance to the pixels that belong to the same region of the image of the pixel $\hat{x}(\boldsymbol{n})$ with respect to the others. A simple and effective tool to do it is given by the bilateral filters [134]. The impulse response of a bilateral filter is

$$w(\boldsymbol{n}, \boldsymbol{\ell}) = \frac{1}{h(\boldsymbol{n})} w_s(\boldsymbol{n}, \boldsymbol{\ell}) w_r(\boldsymbol{n}, \boldsymbol{\ell}), \tag{7.25}$$

where $h(\boldsymbol{n}) = \sum_{\boldsymbol{\ell}} w_s(\boldsymbol{n}, \boldsymbol{\ell}) w_r(\boldsymbol{n}, \boldsymbol{\ell})$ is a normalization constant, $w_s(\boldsymbol{n}, \boldsymbol{\ell})$ is based on *geometric* (spatial in case of images) information and $w_r(\boldsymbol{n}, \boldsymbol{\ell})$ considers the *radiometric* distance between the intensity values of the locations $\boldsymbol{n}$ and $\boldsymbol{\ell}$ [134, 41].

We apply the bilateral filter for the estimation of the HR image. In particular we set

- $w_s(\boldsymbol{n}, \boldsymbol{\ell}) = g(\boldsymbol{n}, \boldsymbol{\ell})$, with the filter coefficients computed as in (7.8) using the proposed optimal procedure;
- $w_r(\boldsymbol{n}, \boldsymbol{\ell}) = \exp\left\{-\frac{(\tilde{x}(\boldsymbol{n}) - x(\boldsymbol{\ell}))^2}{\sigma_r^2}\right\}$, where σ_r^2 is a parameter chosen on the basis of the desired amount of combination of pixel values, as proposed in [134, 41], and $\tilde{x}(\boldsymbol{n})$ is an approximation of the unknown value $x(\boldsymbol{n})$, obtained, for instance, with the bilinear interpolation.

(a) baboon (b) barbara (c) cameraman

(d) lena (e) peppers

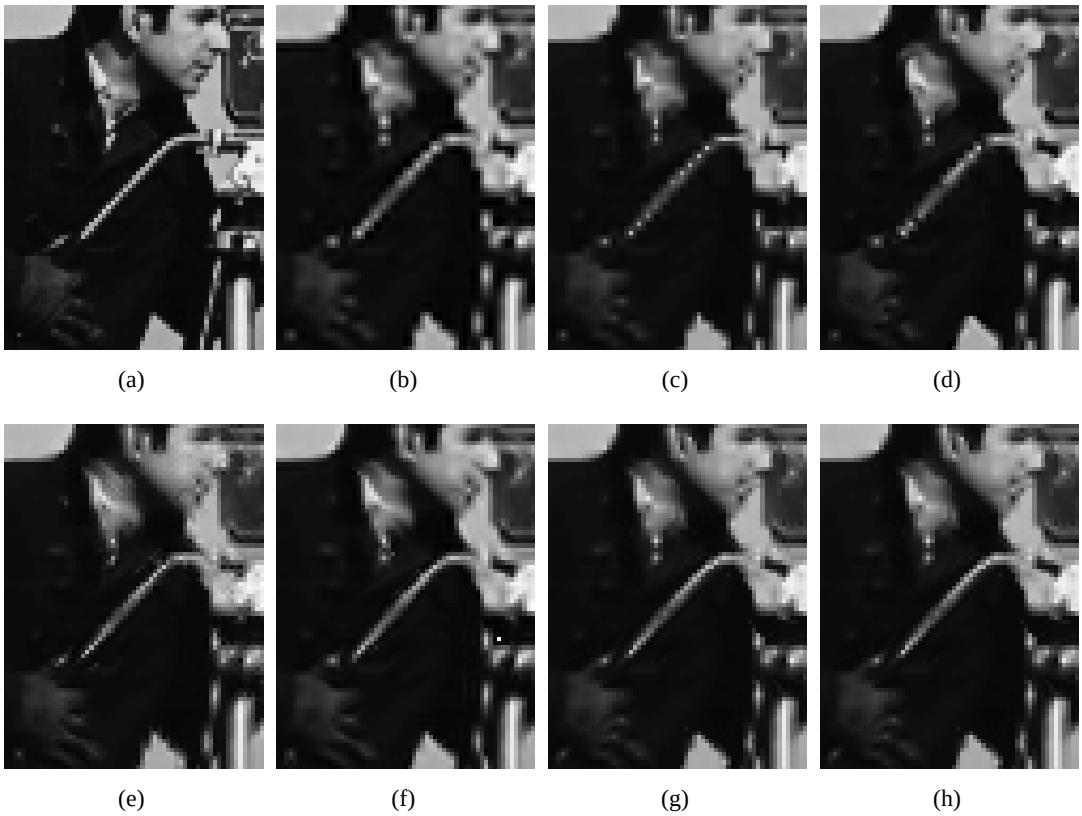
Figure 7.3: *The five test images used in the experimental section.*

The proposed bilateral filter mantains the adaptivity to the local characteristics of image as the filter $g(\boldsymbol{n}, \boldsymbol{\ell})$ and tends to gives better relevance to the neighboring pixels which intensity values are closer to $\tilde{x}(\boldsymbol{n})$.

7.5 Experimental Results

In order to evaluate the performances of the proposed approach, we compare it with some existing image interpolation methods. We use the five test images shown in Fig. 7.3. The image *cameraman* has a resolution of 256×256, while the other images have size 512×512. We downsampled them by a factor z both horizontally and vertically, added a gaussian noise as in Fig. 7.1, then we estimated the high-resolution images using the bilinear interpolation, the bicubic convolution interpolation [78], the techniques described in [21, 141, 85, 147], and the proposed approach. Finally we computed the PSNR values between the original images and their reconstructed versions.

Table 7.1 reports the PSNR values obtained when $z = 2$, that is when the resolution is doubled both horizontally and vertically, and the noise is not added. This is the same configuration applied in [141, 85, 147]. The proposed approach is tested using two different sizes for the support of the reconstruction filter. In the first case the support is defined into a 5×5 window. It corresponds to have four non-null coefficients in the pixels that have four neighboring pixels belonging to the LR image placed along the diagonal directions (such as the pixel labeled with



(a) (b) (c) (d)

(e) (f) (g) (h)

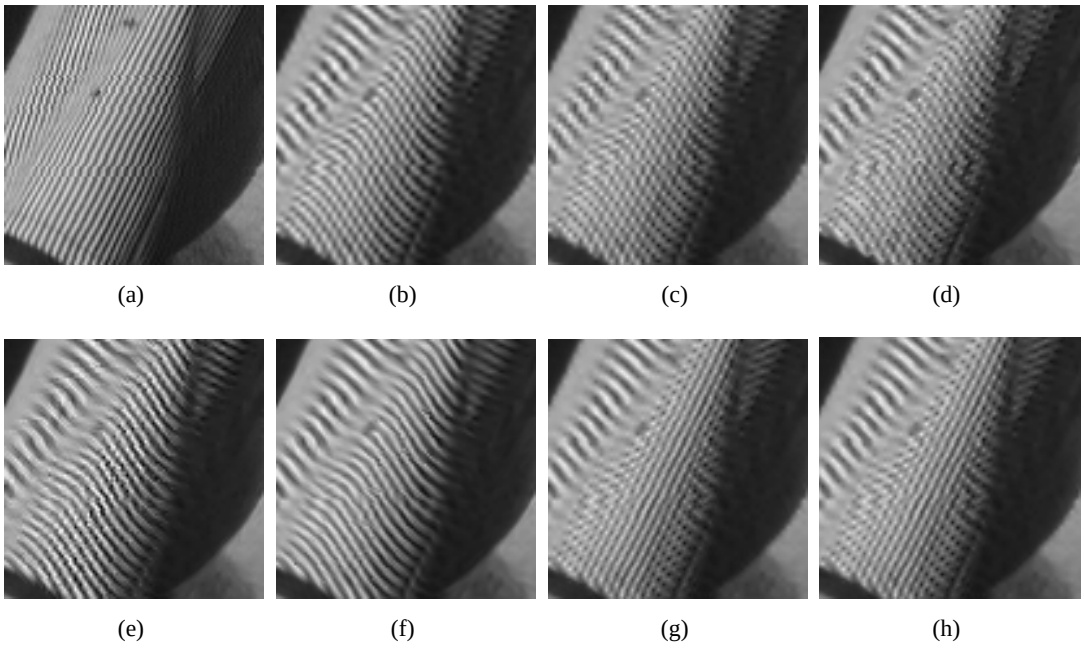
Figure 7.4: *Portion of the image cameraman sampled by factor* 2 *and interpolated by different methods: (a) original image, (b) bicubic interpolation [78], (c) technique [21], (d) technique [141], (e) technique [85], (f) technique [147], (g) proposed approach, (h) proposed approach with the bilateral filter.*

Images	Bilinear Interp.	Bicubic Convol.	Method [21]	Method [141]	Method [85]	Method [147]	Optimal Filter 5×5	Optimal Filter 7×7	Bilateral Filter 5×5	Bilateral Filter 7×7
Baboon	24.00	23.79	23.99	23.71	24.09	23.97	24.01	23.96	24.02	23.99
Barbara	25.15	24.65	25.34	24.84	22.12	23.36	25.05	25.04	24.98	25.02
Cameraman	24.59	24.54	24.54	24.79	24.74	24.67	24.85	24.99	24.89	25.02
Lena	33.45	33.99	33.56	33.97	33.92	34.70	33.56	33.87	33.77	33.92
Peppers	32.81	32.90	32.93	33.14	33.26	33.14	32.88	33.01	33.00	33.05
Average	**28.00**	**27.97**	**28.07**	**28.09**	**27.63**	**27.97**	**28.07**	**28.17**	**28.13**	**28.20**

Table 7.1: *PSNR comparison* (dB) *of different image interpolation approaches for* $z = 2$.

C in Fig. 7.2), and six coefficients for the locations where the neighboring known pixels are placed horizontally or vertically. In the second case the support of the filter is defined into a 7×7 window, and the filter has 16 non-zero coefficients in the first type of pixels and 12 elsewhere. Moreover, the proposed approach is applied also using the improvement based on a bilateral filter that is described in Section 7.4.1 (setting $\sigma_r = 3000$).

The proposed approach outperforms the other techniques when using a 7×7 filter, but also with a 5×5 filter support the performances are good. In particular, the proposed estimation performs well with the most detailed images, such as *baboon*, *barbara* and *cameraman*, since the adaptive approach is able to detect the local characteristics of the image and drive the reconstruction according to them. The use of the bilateral filter gives a slight improvement of



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 7.5: *Portion of the image barbara sampled by factor* 2 *and interpolated by different methods: (a) original image, (b) bicubic interpolation [78], (c) technique [21], (d) technique [141], (e) technique [85], (f) technique [147], (g) proposed approach, (h) proposed approach with the bilateral filter.*

Images	Bilinear Interp.	Bicubic Convol.	Method [130]	Optimal Filter	Bilateral Filter
Baboon	22.22	21.86	21.84	22.17	22.18
Barbara	22.77	22.37	22.67	22.76	22.69
Cameraman	21.72	21.53	21.37	21.79	21.91
Lena	29.95	30.21	30.63	30.24	30.36
Peppers	29.94	29.94	29.83	30.15	30.24
Average	**25.32**	**25.18**	**25.27**	**25.42**	**25.48**

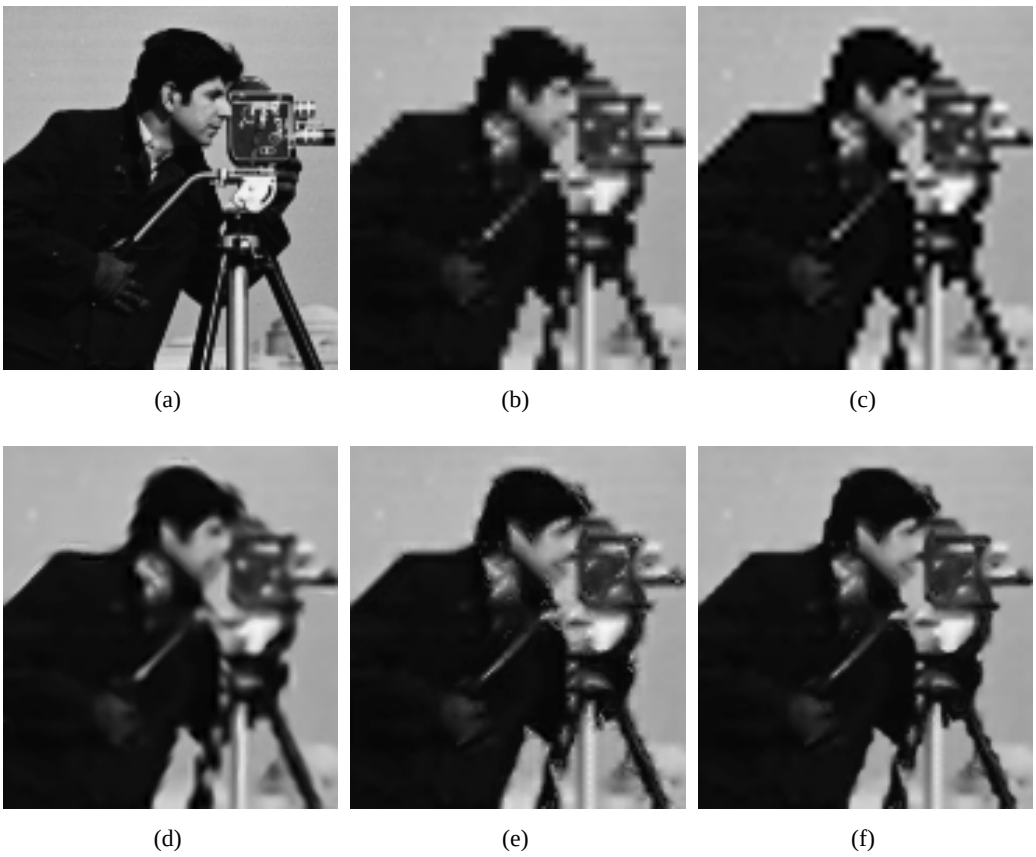
Table 7.2: *PSNR comparison* (dB) *of different image interpolation approaches for* $z = 3$.

the performance in term of PSNR.

The quality of the estimated images is better analyzed with a visual comparison. Fig. 7.4 shows a sample of the test image *cameraman* and its reconstructions (after a sampling with $z = 2$) obtained with the methods described in [78, 21, 141, 85, 147] and with the proposed approach. The bicubic interpolation produces blurred edges and smooth transitions between different intensity regions. These artifacts are reduced by the adaptive techniques; however, many of them introduce aliasing near the edges, as in Fig. 7.4(e). The best compromise between sharpness of the edges and no staircasing effect is given by the proposed approach, that reconstructs well also the diagonal details.

Fig. 7.5 shows a sample of the test image *barbara* including some high-frequency details. Many approaches fail to reconstruct them and are not able to detect their orientation. Instead, the proposed approach improves the estimation of the correct directions of the strips and reduces the aliasing effects.

The improvement given by the bilateral filter with respect to the optimal filter is more ev-



(a) (b) (c)

(d) (e) (f)

Figure 7.6: *Portion of the image cameraman sampled by factor* 3 *and interpolated by different methods: (a) original image, (b) bilinear interpolation, (c) bicubic interpolation [78], (d) technique [130], (e) proposed approach, (f) proposed approach with the bilateral filter.*

ident in the visual quality of the reconstructed images when the sampling factor is $z = 3$. Table 7.2 reports a comparison between different interpolation methods with the scheme of Fig. 7.1 and $z = 3$. The PSNR values obtained using the bilinear interpolation, the bicubic interpolation, the method described in [130], the proposed optimal approach (with a 5×5 filter support), and the proposed approach with bilateral filter (with the same filter support and $\sigma_r = 6000$) are compared. The images reconstructed with the proposed strategies have a better quality than the images interpolated with the non-adaptive methods and with the technique [130]. This can be observed also in Fig. 7.6 where the portion of the image *cameraman* reconstructed with the proposed approach presents less zigzagging edges and sharper details with respect to the image obtained with the cubic interpolation. Moreover, the bilateral filter removes the moiré artifacts introduced by the optimal filter and improves the sharpness of the edges.

To check the robustness of the different interpolation methods to the noise effects, we try to improve the resolution when the LR image (sampled with $z = 2$, both horizontally and vertically) is corrupted with a gaussian noise with standard deviation $\sigma_\eta = 5$. Table 7.3 reports the performances in PSNR of the same methods analyzed in Table 7.1. We observe that the proposed parameters estimation is not affected by the noise and the quality of the reconstructed images does not change dramatically. Moreover, the proposed technique is more robust than

Images	Bilinear Interp.	Bicubic Convol.	Method [21]	Method [85]	Method [141]	Method [147]	Optimal Filter 5×5	Optimal Filter 7×7	Bilateral Filter 5×5	Bilateral Filter 7×7
Baboon	23.77	23.54	23.76	23.46	23.86	23.69	23.79	23.74	23.81	23.77
Barbara	24.85	24.33	25.02	24.52	22.11	23.22	24.80	24.78	24.73	24.77
Cameraman	24.32	24.23	24.27	24.50	24.47	24.34	24.65	24.76	24.64	24.73
Lena	31.75	31.82	31.81	31.92	32.05	32.28	32.04	32.17	32.17	32.22
Peppers	31.29	31.14	31.38	31.40	31.61	31.27	31.60	31.64	31.67	31.67
Average	**27.20**	**27.01**	**27.25**	**27.16**	**26.82**	**26.96**	**27.38**	**27.42**	**27.40**	**27.43**

Table 7.3: *PSNR comparison* (dB) *of different image interpolation approaches for* $z = 2$ *and* $\sigma_\eta = 5$.

Images	Bilinear Interp.	Bicubic Convol.	Method [21]	Method [85]	Method [141]	Method [147]	Optimal Filter 5×5	Optimal Filter 7×7	Bilateral Filter 5×5	Bilateral Filter 7×7
Baboon	24.28	24.19	23.98	24.05	24.39	24.38	24.25	24.23	24.28	24.26
Barbara	25.35	24.99	25.52	25.08	22.60	23.83	25.36	25.35	25.31	25.34
Cameraman	24.82	24.88	24.46	25.09	25.02	25.09	25.03	25.19	25.11	25.25
Lena	33.51	34.22	33.41	34.15	34.08	34.92	33.53	33.90	33.75	33.93
Peppers	33.10	33.32	33.43	33.51	33.59	33.56	33.10	33.28	33.23	33.30
Average	**28.21**	**28.32**	**28.16**	**28.38**	**27.94**	**28.36**	**28.25**	**28.39**	**28.34**	**28.42**

Table 7.4: *PSNR comparison* (dB) *of different image interpolation approaches for* $z = 2$. *A low-pass gaussian prefilter with* $\sigma_g = 0.4$ *is applied prior to sampling.*

the other adaptive approaches.

In Section 7.2 we assumed that the LR image is generated by directly sampling the corresponding HR image, as in Fig 7.1. An alternative model considers a low-pass filtering prior to sampling, in order to reduce the effects of the aliasing. To show the effectiveness of the proposed algorithm also in this situation, Table 7.4 reports the performances of different techniques when a gaussian low-pass filter with standard deviation $\sigma_g = 0.4$ is applied before the sampling. The PSNR is computed between the original image (prior to filtering) and the resulting estimates. In this case the performances of the non-adaptive interpolation methods, such as the cubic convolution, are superior with respect to the sampling without prefilter, because the sampled images are smoother and less aliased. However, the PSNR results show that the proposed adaptive approach outperforms the simplest strategies also in this configuration.

Related Publications

[p1] D. Menon, S. Andriani, G. Calvagno, and T. Erseghe, “On the dependency between compression and demosaicing in digital cinema,” in *Proc. of Conference on Visual Media Production*", Nov. 2005, pp 104-111.

[p2] D. Menon, S. Andriani, and G. Calvagno, “A novel technique for reducing demosaicing artifacts,” in *Proc. of the European Signal Processing Conference (EUSIPCO)*, Sept. 2006.

[p3] S. Andriani, G. Calvagno and D. Menon, “Lossless compression of Bayer mask images using an optimal vector prediction technique,” in *Proc. of the European Signal Processing Conference (EUSIPCO)*, Sept. 2006.

[p4] D. Menon, S. Andriani, and G. Calvagno, “Demosaicing with directional filtering and *a posteriori* decision,” *IEEE Trans. Image Processing*, vol. 16, no. 1, pp.132–14, Jan. 2007.

[p5] D. Menon and G. Calvagno, “Demosaicing based on wavelet analysis of the luminance component,” in *Proc. IEEE Int. Conf. Image Processing*, vol. 2, Sept. 2007, pp. 181–184.

[p6] D. Menon and G. Calvagno, “A regularization approach to demosaicking,” in *Proc. SPIE-IS&T Electronic Imaging, Visual Communications and Image Processing*, vol. 6822, Jan. 2008.

[p7] D. Menon, L. Sbaiz, and G. Calvagno, “Optimal image interpolation based on local statistics,” submitted for publication to *IEEE Trans. Image Processing*, 2008.

[p8] D. Menon and G. Calvagno, “Regularization approaches to demosaicking,” submitted for publication to *IEEE Trans. Image Processing*, 2008.

Bibliography

[1] J. E. Adams, “Interactions between color plane interpolation and other image processing functions in electronic photograpy,” in *Proc. SPIE*, vol. 2416, Feb. 1995, pp. 144–151.

[2] ——, “Design of practical color filter array interpolation algorithms for digital cameras,” in *Proc. SPIE*, vol. 3028, Feb. 1997, pp. 117–125.

[3] ——, “Design of practical color filter array interpolation algorithms for digital cameras, part 2,” in *Proc. IEEE Int. Conf. Image Processing*, vol. 1, Oct. 1998, pp. 488–492.

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

[5] J. Allebach and P. W. Wong, “Edge-directed interpolation,” in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, Sept. 1996, pp. 707–710.

[6] D. Alleysson, S. Süsstrunk, and J. Hérault, “Linear demosaicing inspired by the human visual system,” *IEEE Trans. Image Processing*, vol. 14, no. 4, pp. 439–449, Apr. 2005.

[7] D. Alleysson, S. Süsstrunk, and J. Marguier, “Influence of spectral sensitivity functions on color demosaicing,” in *IS&T/SID 11th Color Imaging Conference*, vol. 11, 2003, pp. 351–357.

[8] H. A. Aly and E. Dubois, “Image up-sampling using total-variation regularization with a new observation model,” *IEEE Trans. Image Processing*, vol. 14, no. 10, pp. 1647–1659, Oct. 2005.

[9] M. R. Banham and A. K. Katsaggelos, “Digital image restoration,” *IEEE Signal Processing Mag.*, vol. 14, no. 2, pp. 24–41, Mar. 1997.

[10] K. Barnard, V. Cardei, and B. Funt, “A comparison of computational color constancy algorithms - part I: Methodology and experiments with synthesized data,” *IEEE Trans. Image Processing*, vol. 11, no. 9, pp. 972–983, Sept. 2002.

[11] S. Battiato, M. Guarnera, G. Messina, and V. Tomaselli, “Recent patents on color demosaicing,” *Recent Patents on Computer Science*, vol. 1, no. 3, pp. 94–207, Nov. 2008.

[12] B. E. Bayer, “Color imaging array,” U.S. Patent 3 971 065, July, 1976.

[13] M. Belge, M. E. Kilmer, and E. L. Miller, "Efficient determination of multiple regularization parameters in a generalized L-curve framework," *Inverse Problems*, vol. 18, no. 4, pp. 1161–1183, July 2002.

[14] M. Bertero, T. A. Poggio, and V. Torre, "Ill-posed problems in early vision," *Proceedings of the IEEE*, vol. 76, no. 8, pp. 869–889, Aug. 1988.

[15] J. M. Bioucas-Dias, M. A. T. Figueiredo, and J. P. Oliveira, "Total variation-based image deconvolution: a majorization-minimization approach," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, vol. II, 2006, pp. 861–864.

[16] D. H. Brainard, "Bayesan method for reconstructing color images from thrichromatic samples," in *Proc. IS&T 47th Annual Meeting*, 1994, pp. 375–380.

[17] A. Buades, B. Coll, and J. M. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Modeling Simulation*, vol. 4, no. 2, pp. 490–530, July 2005.

[18] G. Calvagno, G. A. Mian, and R. Rinaldo, "Synthesis filter bank optimization in two-dimensional separable subband coding systems," *IEEE Trans. Image Processing*, vol. 9, no. 9, pp. 1497–1508, Sept. 2000.

[19] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 8, pp. 679–698, 1986.

[20] L. Cappellari and G. A. Mian, "Analysis of joint predictive-transform coding for still image compression," *Signal Processing*, vol. 84, no. 11, pp. 2097–2114, Nov. 2004.

[21] S. Carrato, G. Ramponi, and S. Marsi, "A simple edge-sensitive image interpolation filter," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, Sept. 1996, pp. 711–714.

[22] B. Chaix de Lavarène, D. Alleysson, B. Durette, and J. Hérault, "Efficient demosaicing through recursive filtering," in *Proc. IEEE Int. Conf. Image Processing*, vol. 2, Sept. 2007, pp. 189–192.

[23] B. Chaix de Lavarène, D. Alleysson, and J. Hérault, "Practical implementation of LMMSE demosaicing using luminance and chrominance spaces," *Computer Vision and Image Understanding, Special issue on Color Image Processing*, vol. 107, pp. 3–13, July-Aug. 2007.

[24] C.-L. Chang and B. Girod, "Direction-adaptive discrete wavelet transform for image compression," *IEEE Trans. Image Processing*, vol. 16, no. 5, May 2007.

[25] E. Chang, S. Cheung, and D. Y. Pan, "Color filter array recovery using a threshold-based variable number of gradients," in *Proc. SPIE*, vol. 3650, 1999, pp. 36–43.

[26] H.-A. Chang and H. H. Chen, "Stochastic color interpolation for digital cameras," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 8, pp. 964–973, Aug. 2007.

[27] S. G. Chang, B. Yu, and M. Vetterli, "Adaptive wavelet thresholding for image denoising and compression," *IEEE Trans. Image Processing*, vol. 9, no. 9, pp. 1532–1546, Sept. 2000.

[28] S. G. Chang, Z. Cvetković, and M. Vetterli, "Locally adaptive wavelet-based image interpolation," *IEEE Trans. Image Processing*, vol. 15, no. 6, pp. 1471–1485, June 2006.

[29] P. Charbonnier, L. Blanc-Féraud, G. Aubert, and M. Barlaud, "Deterministic edge-preserving regularization in computed imaging," *IEEE Trans. Image Processing*, vol. 6, no. 2, pp. 298–311, Feb. 1997.

[30] L. Chen, K.-H. Yap, and Y. He, "Color filter array demosaicking using wavelet-based subband synthesis," in *Proc. IEEE Int. Conf. Image Processing*, vol. 2, Sept. 2005, pp. 1002–1005.

[31] K.-H. Chung and Y.-H. Chan, "Color demosaicing using variance of color differences," *IEEE Trans. Image Processing*, vol. 15, no. 10, pp. 2944–2955, Oct. 2006.

[32] K. H. Chung and Y. H. Chan, "A low-complexity joint color demosaicking and zooming algorithm for digital camera," *IEEE Trans. Image Processing*, vol. 16, no. 7, pp. 1705–1715, July 2007.

[33] D. R. Cok, "Signal processing method and apparatus for producing interpolated chrominance values in a sampled color image signal," U.S. Patent 4 642 678, Feb., 1987.

[34] G. Demoment, "Image reconstruction and restoration: overview of common estimation structures and problems," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, no. 12, pp. 2024–2036, Dec. 1989.

[35] D. L. Donoho, "De-noising by soft-thresholding," *IEEE Trans. Inform. Theory*, vol. 41, no. 3, pp. 613–627, May 1995.

[36] D. L. Donoho and I. M. Johnstone, "Ideal spatial adaptation via wavelet shrinkage," *Biometrika*, vol. 81, pp. 425–455, 1994.

[37] J. Driesen and P. Scheunders, "Wavelet-based color filter array demosaicking," in *Proc. IEEE Int. Conf. Image Processing*, vol. 5, Oct. 2004, pp. 3311–3314.

[38] E. Dubois, "The sampling and reconstruction of time-varying imagery with application in video systems," *Proceedings of the IEEE*, vol. 73, no. 4, pp. 502–522, Apr. 1985.

[39] ——, "Frequency-domain methods for demosaicking of bayer-sampled color images," *IEEE Signal Processing Lett.*, vol. 12, no. 12, pp. 847–850, Dec. 2005.

[40] ——, "Filter design for adaptive frequency-domain bayer demosaicking," in *Proc. IEEE Int. Conf. Image Processing*, Oct. 2006, pp. 2705–2708.

[41] M. Elad, "On the origin of the bilateral filter and ways to improve it," *IEEE Trans. Image Processing*, vol. 11, no. 10, pp. 1141–1151, Oct. 2002.

[42] ——, "Why simple shrinkage is still relevant for redundant representations?" *IEEE Trans. Inform. Theory*, vol. 52, no. 12, pp. 5559–5569, Dec. 2006.

[43] S. Farsiu, M. Elad, and P. Milanfar, "Multiframe demosaicing and super-resolution of color images," *IEEE Trans. Image Processing*, vol. 15, no. 1, pp. 141–159, Jan. 2006.

[44] S. Farsiu, M. D. Robinson, M. Elad, and P. Milanfar, "Fast and robust multiframe super resolution," *IEEE Trans. Image Processing*, vol. 13, no. 10, pp. 1327–1344, Oct. 2004.

[45] A. Foi, M. Trimeche, V. Katkovnik, and K. Egiazarian, "Practical poissonian-gaussian noise modeling and fitting for single-image raw-data," *IEEE Trans. Image Processing*, vol. 17, no. 10, pp. 1737–1754, Oct. 2008.

[46] W. T. Freeman, "Method and apparatus for reconstructing missing color samples," U.S. Patent 4 774 565, Sept., 1988.

[47] D. Geman and G. Reynolds, "Constrained restoration and the recovery of discontinuities," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 14, no. 3, pp. 367–383, Mar. 1992.

[48] D. Geman and C. Yang, "Nonlinear image recovery with half-quadratic regularization," *IEEE Trans. Image Processing*, vol. 4, no. 7, pp. 932–946, July 1995.

[49] J. W. Glotzbach, R. W. Schafer, and K. Illgner, "A method of color filter array interpolation with alias cancellation properties," in *Proc. IEEE Int. Conf. Image Processing*, vol. 1, 2001, pp. 141–144.

[50] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd ed. Englewood Cliffs, NJ: Prentice Hall, 2002.

[51] T. Gotoh and M. Okutomi, "Direct super-resolution and registration using raw CFA images," in *Proc. Int. Conf. Computer Vision and Pattern Recognition*, vol. 2, July 2004, pp. 600–607.

[52] B. K. Gunturk, Y. Altunbasak, and R. M. Mersereau, "Color plane interpolation using alternating projections," *IEEE Trans. Image Processing*, vol. 11, no. 9, pp. 997–1013, Sept. 2002.

[53] B. K. Gunturk, J. Glotzbach, Y. Altunbasak, R. W. Schafer, and R. M. Mersereau, "Demosaicking: color filter array interpolation," *IEEE Signal Proc. Mag.*, vol. 22, no. 1, pp. 44–54, Jan. 2005.

[54] M. R. Gupta and T. Chen, "Vector color filter array demosaicing," in *Proc. SPIE Conference on Sensors and Camera Systems for Scientific, Industrial, and Digital Photography Applications II*, vol. 4306, May 2001, pp. 374–382.

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

[56] J. F. Hamilton and J. T. Compton, "Processing color and panchromatic pixels," U.S. Patent 2007 0 024 879, Feb., 2007.

[57] M. H. Hayes, *Statistical Digital Signal Processing and Modeling*. New York, NY: John Wiley & Sons, 1996.

[58] Y. Hel-Or, "The canonical correlations of color images and their use for demosaicing," HP Labs Israel, Tech. Rep. HPL-2003-164R1, Feb. 2004.

[59] Y. Hel-Or and D. Keren, "Demosaicing of color images using steerable wavelets," HP Labs Israel, Tech. Rep. HPL-2002-206R1, Aug. 2002.

[60] R. H. Hibbard, "Apparatus and method for adaptively interpolating a full color image utilizing chrominance gradients," U.S. Patent 5 382 976, Jan., 1995.

[61] K. Hirakawa, "Fourier and filterbank analyses of signal-dependent noise," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, Apr. 2008, pp. 3517–3520.

[62] K. Hirakawa and X.-L. Meng, "An empirical Bayes EM-wavelet unification for simultaneous denoising, interpolation, and/or demosaicing," in *Proc. IEEE Int. Conf. Image Processing*, Oct. 2006, pp. 1453–1456.

[63] K. Hirakawa, X.-L. Meng, and P. J. Wolfe, "A framework for wavelet-based analysis and processing of color filter array images with applications to denoising and demosaicking," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, vol. I, Apr. 2007, pp. 597–600.

[64] K. Hirakawa and T. W. Parks, "Adaptive homogeneity-directed demosaicing algorithm," *IEEE Trans. Image Processing*, vol. 14, no. 3, pp. 360–369, Mar. 2005.

[65] ——, "Joint demosaicing and denoising," *IEEE Trans. Image Processing*, vol. 15, no. 8, pp. 2146–2157, Aug. 2006.

[66] K. Hirakawa and P. J. Wolfe, "Spatio-spectral color filter array design for enhanced image fidelity," in *Proc. IEEE Int. Conf. Image Processing*, vol. 2, Sept. 2007, pp. 81–84.

[67] ——, "Advancing the digital camera pipeline for mobile multimedia: key challenges from a signal processing perspective," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, Apr. 2008, pp. 5332–5335.

[68] ——, "Second-generation color filter array and demosaicking designs," in *Proc. SPIE-IS&T Electronic Imaging, Visual Communications and Image Processing*, vol. 6822, Jan. 2008.

[69] ——, "Spatio-spectral color filter array design for optimal image recovery," *IEEE Trans. Image Processing*, vol. 17, no. 10, pp. 1876–1890, Oct. 2008.

[70] J. Idier, "Convex half-quadratic criteria and interacting auxiliary variables for image restoration," *IEEE Trans. Image Processing*, vol. 10, no. 7, pp. 1001–1009, July 2001.

[71] A. K. Jain, *Fundamentals of digital image processing*. Englewood Cliffs, NJ: Prentice Hall, 1989.

[72] R. Kakarala and Z. Baharav, "Adaptive demosaicing with the principal vector method," *IEEE Trans. Consumer Electron.*, vol. 48, no. 4, pp. 932–937, Nov. 2002.

[73] O. Kalevo and H. Rantanen, "Noise reduction techniques for Bayer-matrix images," in *Proc. SPIE Sensor and Camera systems for scientific, industrial and digital photography applications III*, vol. 4669, 2002.

[74] W. C. Karl, "Regularization in image restoration and reconstruction," in *Handbook of Image and Video Processing*, A. Bovik, Ed. San Diego, CA: Academic, 2000, ch. 3.6, pp. 141–160.

[75] A. K. Katsaggelos, J. Biemond, R. W. Schafer, and R. M. Mersereau, "A regularized iterative image restoration algorithm," *IEEE Trans. Signal Processing*, vol. 39, no. 4, pp. 914–929, Apr. 1991.

[76] A. K. Katsaggelos and C.-J. Tsai, "Iterative image restoration," in *Handbook of Image and Video Processing*, A. Bovik, Ed. San Diego, CA: Academic, 2000, ch. 3.9, pp. 191–206.

[77] D. Keren and M. Osadchy, "Restoring subsampled color images," *Machine Vision and Applications*, vol. 11, no. 4, pp. 197–202, Dec. 1999.

[78] R. G. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 29, no. 6, pp. 1153–1160, Dec. 1981.

[79] R. Kimmel, "Demosaicing: Image reconstruction from ccd samples," *IEEE Trans. Image Processing*, vol. 8, no. 9, pp. 1221–1228, Sept. 1999.

[80] C. A. Laroche and M. A. Prescott, "Apparatus and method for adaptively interpolating a full color image utilizing chrominance gradients," U.S. Patent 5 373 322, Dec., 1994.

[81] A. Levin, R. Fergus, F. Durand, and W. T. Freeman, "Image and depth from a conventional camera with a coded aperture," *ACM Transactions on Graphics, SIGGRAPH 2007 Conference Proceedings*, vol. 26, no. 3, July 2007.

[82] M. Li and T. Q. Nguyen, "Markov random field model-based edge-directed image interpolation," *IEEE Trans. Image Processing*, vol. 17, no. 7, pp. 1121–1128, July 2008.

[83] X. Li, "Demosaicing by successive approximation," *IEEE Trans. Image Processing*, vol. 14, no. 3, pp. 370–379, Mar. 2005.

[84] X. Li, B. K. Gunturk, and L. Zhang, "Image demosaicing: a systematic survey," in *Proc. SPIE-IS&T Electronic Imaging, Visual Communications and Image Processing*, vol. 6822, no. 1, Jan. 2008.

[85] X. Li and M. Orchard, "New edge-directed interpolation," *IEEE Trans. Image Processing*, vol. 10, no. 10, pp. 1521–1527, Oct. 2001.

[86] Y. Li and F. Santosa, "A computational algorithm for minimizing total variation in image restoration," *IEEE Trans. Image Processing*, vol. 5, no. 6, pp. 987–995, June 1996.

[87] N.-X. Lian, L. Chang, Y.-P. Tan, and V. Zagorodnov, "Adaptive filtering for color filter array demosaicking," *IEEE Trans. Image Processing*, vol. 16, no. 10, pp. 2515 – 2525, Oct. 2007.

[88] 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, Jan. 2002.

[89] W. Lu and Y.-P. Tan, "Color filter array demosaicing: New method and performance measures for color filter array," *IEEE Trans. Image Processing*, vol. 12, no. 10, pp. 1194–1210, Oct. 2003.

[90] R. Lukac, K. Martin, and K. Plataniotis, "Digital camera zooming based on unified cfa image processing steps," *IEEE Trans. Consumer Electron.*, vol. 50, no. 1, pp. 15–24, Feb. 2004.

[91] R. Lukac, K. N. Plataniotis, and D. Hatzinakos, "Color image zooming on the bayer pattern," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 11, 2005.

[92] R. Lukac, Ed., *Single-Sensor Imaging: Methods and applications for digital cameras*. Atlanta, GA: CRC Press, 2008.

[93] R. Lukac and K. N. Plataniotis, "Color filter arrays: Design and performance analysis," *IEEE Trans. Consumer Electron.*, vol. 51, no. 4, pp. 1260–1267, Nov. 2005.

[94] ——, "Universal demosaicking for imaging pipelines with an RGB color filter array," *Pattern Recognition*, vol. 38, no. 11, pp. 2208–2212, Nov. 2005.

[95] R. Lukac, K. N. Plataniotis, D. Hatzinakos, and M. Aleksic, "A novel cost effective demosaicing approach," *IEEE Trans. Consumer Electron.*, vol. 50, no. 1, pp. 256–261, Feb. 2004.

[96] ——, "A new CFA interpolation framework," *Signal Processing*, vol. 86, no. 7, pp. 1559–1579, July 2006.

[97] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Trans. Image Processing*, vol. 17, no. 1, pp. 53–69, Jan. 2008.

[98] S. Mallat and W. L. Hwang, "Singularity detection and processing with wavelets," *IEEE Trans. Inform. Theory*, vol. 38, no. 2, pp. 617–643, Mar. 1992.

[99] S. Mallat and S. Zhong, "Characterization of signals from multiscale edges," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 14, no. 7, pp. 710–732, Jan. 1992.

[100] L. Meylan, D. Alleysson, and S. Süsstrunk, "A model of retinal local adaptation for the tone mapping of color filter array images," *Journal of the Optical Society of America A (JOSA A)*, vol. 24, no. 9, pp. 2807–2816, 2007.

[101] N. Mueller, Y. Lu, and M. N. Do, "Image interpolation using multiscale geometric representations," in *SPIE Conf. on Computational Imaging V*, vol. 6498, no. 1, Jan. 2007.

[102] J. Mukherjee, R. Parthasarathi, and S. Goyal, "Markov random field processing for color demosaicing," *Pattern Recognition Letters*, vol. 22, no. 3-4, pp. 339–351, Mar. 2001.

[103] D. D. Muresan and T. W. Parks, "Demosaicing using optimal recovery," *IEEE Trans. Image Processing*, vol. 14, no. 2, pp. 267–278, Feb. 2005.

[104] M. K. Ng, H. Shen, E. Y. Lam, and L. Zhang, "A total variation regularization based super-resolution reconstruction algorithm for digital video," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, 2007.

[105] M. Nikolova, "Analytical bounds on the minimizers of (nonconvex) regularized least-squares," *SIAM Journal on Inverse Problems and Imaging*, vol. 2, no. 1, pp. 133–149, Feb. 2008.

[106] M. Nikolova and R. H. Chan, "The equivalence of half-quadratic minimization and the gradient linearization iteration," *IEEE Trans. Image Processing*, vol. 16, no. 6, pp. 1623–1627, June 2007.

[107] M. Nikolova and M. K. Ng, "Analysis of half-quadratic minimization methods for signal and image recovery," *SIAM Journal on Scientific Computing*, vol. 27, no. 3, pp. 937–966, 2005.

[108] D. Paliy, A. Foi, R. Bilcu, and V. Katkovnik, "Denoising and interpolation of noisy Bayer data with adaptive cross-color filters," in *Proc. SPIE-IS&T Electronic Imaging, Visual Communications and Image Processing*, vol. 6822, Jan. 2008.

[109] D. Paliy, V. Katkovnik, R. Bilcu, S. Alenius, and K. Egiazarian, "Spatially adaptive color filter array interpolation for noiseless and noisy data," *International Journal of Imaging Systems and Technology, Special Issue on Applied Color Image Processing*, vol. 17, no. 3, pp. 105–122, Oct. 2007.

[110] S. C. Park, M. K. Park, and M. G. Kang, "Super-resolution image reconstruction: a technical overview," *IEEE Signal Processing Mag.*, vol. 20, no. 3, pp. 21–36, May 2003.

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

[112] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 12, no. 7, pp. 629–639, July 1990.

[113] T. Q. Pham, L. J. van Vliet, and K. Schutte, "Robust fusion of irregularly sampled data using adaptive normalized convolution," *EURASIP Journal on Applied Signal Processing*, vol. 2006, article ID 83268, 12 pages, 2006.

[114] A. Pižurica and W. Philips, "Estimating the probability of the presence of a signal of interest in multiresolution single- and multiband image denoising," *IEEE Trans. Image Processing*, vol. 15, no. 3, pp. 654–665, Mar. 2006.

[115] T. A. Poggio, V. Torre, and C. Koch, "Computational vision and regularization theory," *Nature*, vol. 317, pp. 314 – 319, Sept. 1985.

[116] A. B. Poirson and B. A. Wandell, "Appearance of colored patterns: pattern-color separability," *Journal of the Optical Society of America (JOSA)*, vol. 10, no. 12, pp. 2458–2470, 1993.

[117] ——, "Pattern-color separable pathways predict sensitivity to simple colored patterns," *Vision Research*, vol. 35, no. 2, pp. 239–254, 1996.

[118] J. Portilla, D. Otaduy, and C. Dorronsoro, "Low-complexity linear demosaicing using joint spatial-chromatic image statistics," in *Proc. IEEE Int. Conf. Image Processing*, vol. 1, Sept. 2005, pp. 61–64.

[119] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, "Image denoising using scale mixtures of gaussians in the wavelet domain," *IEEE Trans. Image Processing*, vol. 12, no. 11, pp. 1338–1351, Nov. 2003.

[120] R. Ramanath, W. E. Snyder, Y. Yoo, and M. S. Drew, "Color image processing pipeline," *IEEE Signal Processing Mag.*, vol. 22, no. 1, pp. 34–43, Jan. 2005.

[121] R. Ramanath and W. E. Snyder, "Adaptive demosaicking," *Journal of Electronic Imaging*, vol. 12, no. 4, pp. 633–642, Oct. 2003.

[122] E. Reinhard, M. Stark, P. Shirley, and J. Ferwerda, "Photographic tone reproduction for digital images," in *Proc. of ACM SIGGRAPH, Annual Conference on Computer Graphics*, 2002, pp. 267–276.

[123] E. Reinhard, G. Ward, S. Pattanaik, and P. Debevec, *High Dynamic Range Imaging. Acquisition, Display, and Image-Based Lighting*. Burlington: Morgan Kaufmann, 2005.

[124] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D*, vol. 60, no. 1-4, pp. 259–268, Nov. 1992.

[125] T. Saito and T. Komatsu, "Demosaicing approach based on extended color total-variation regularization," in *Proc. IEEE Int. Conf. Image Processing*, vol. 1, Sept. 2008, pp. 885–888.

[126] ——, "Demosaicing method using the extended color total-variation regularization," in *Proc. IS&T/SPIE Electronic Imaging: Digital Photography III*, vol. 6817, Jan. 2008.

[127] R. W. Schafer, R. M. Mersereau, and M. A. Richards, "Constrained iterative restoration algorithms," *Proceedings of the IEEE*, vol. 69, no. 4, pp. 432–450, Apr. 1981.

[128] R. R. Schultz and R. L. Stevenson, "A bayesian approach to image expansion for improved definition," *IEEE Trans. Image Processing*, vol. 3, no. 3, pp. 233–242, May 1994.

[129] G. Spampinato, A. Bruna, G. Sanguedolce, E. Ardizzone, and M. La Cascia, "Improved color interpolation using discrete wavelet transform," in *Proc. SPIE Image and Video Communications and Processing*, vol. 5685, Jan. 2005, pp. 753–760.

[130] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstruction," *IEEE Trans. Image Processing*, vol. 16, no. 2, pp. 349–366, Feb. 2007.

[131] D. Taubman, "Generalized Wiener reconstruction of images from colour sensor data using a scale inveriant prior," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, 2000, pp. 801–804.

[132] S. Teboul, L. Blanc-Feraud, G. Aubert, and M. Barlaud, "Variational approach for edge-preserving regularization using coupled pde's," *IEEE Trans. Image Processing*, vol. 7, no. 3, pp. 387–397, Mar. 1998.

[133] A. Tikhonov and V. Arsenin, *Solutions of Ill-Posed Problems*. Washington, DC: Winston, 1977.

[134] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. IEEE Int. Conf. Computer Vision*, Jan. 1998, pp. 839–846.

[135] H. J. Trussell and R. E. Hartwig, "Mathematics for demosaicking," *IEEE Trans. Image Processing*, vol. 3, no. 11, pp. 485–492, Apr. 2002.

[136] C.-Y. Tsai and K.-T. Song, "Heterogeneity-projection hard-decision color interpolation using spectral-spatial correlation," *IEEE Trans. Image Processing*, vol. 16, no. 1, pp. 78–91, Jan. 2007.

[137] P. Vandewalle, K. Krichane, D. Alleysson, and S. Süsstrunk, "Joint demosaicing and super-resolution imaging from a set of unregistered aliased images," in *Proc. IS&T/SPIE Electronic Imaging: Digital Photography III*, vol. 6802, Jan. 2007.

[138] C. R. Vogel and M. E. Oman, "Fast, robust total variation-based reconstruction of noisy, blurred images," *IEEE Trans. Image Processing*, vol. 7, no. 6, pp. 813–824, June 1998.

[139] X. Wu, W. K. Choi, and P. Bao, "Color restoration from digital camera data by pattern matching," in *Proc. SPIE*, vol. 3018, 1997, pp. 12–17.

[140] X. Wu and N. Zhang, "Primary-consistent soft-decision color demosaicking for digital cameras (patent pending)," *IEEE Trans. Image Processing*, vol. 13, no. 9, pp. 1263–1274, Sept. 2004.

[141] L. Zhang and X. Wu, "Image interpolation via directional filtering and data fusion," *IEEE Trans. Image Processing*, vol. 15, no. 8, pp. 2226–2238, Aug. 2006.

[142] L. Zhang, R. Lukac, X. Wu, and D. Zhang, "PCA-based spatially adaptive denoising of CFA images for single-sensor digital cameras," *IEEE Trans. Image Processing*, 2009, to be published. [Online]. Available: http://www4.comp.polyu.edu.hk/ cslzhang/PCA_CFA_DN/CFA_Denoising.pdf

[143] L. Zhang and X. Wu, "Color demosaicking via directional linear minimum mean square-error estimation," *IEEE Trans. Image Processing*, vol. 14, no. 12, pp. 2167–2177, Dec. 2005.

[144] L. Zhang, X. Wu, and D. Zhang, "Color reproduction from noisy CFA data of single sensor digital cameras," *IEEE Trans. Image Processing*, vol. 16, no. 9, pp. 2184–2197, Sept. 2007.

[145] L. Zhang and D. Zhang, "A joint demosaicking-zooming scheme for single sensor digital color cameras," *Computer Vision and Image Understanding, Special issue on Color Image Processing*, vol. 107, pp. 14–25, July-Aug. 2007.

[146] X. Zhang and B. A. Wandell, "A spatial extension of cielab for digital color-image reproduction," *Journal of the Society for Information Display*, vol. 5, no. 1, pp. 61–63, 1997.

[147] X. Zhang and X. Wu, "Image interpolation by adaptive 2-d autoregressive modeling and soft-decision estimation," *IEEE Trans. Image Processing*, vol. 17, pp. 887–896, June 2008.