Image quality improvement for digital still cameras : demosaicking and de- ringing

Wang, Xiaomeng

2005

Wang, X.. (2005). Image quality improvement for digital still cameras : demosaicking and de- ringing. Master' s thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/3692>

<https://doi.org/10.32657/10356/3692>

Nanyang Technological University

Downloaded on 26 Apr 2023 18:23:01 SGT

**Image Quality Improvement
for Digital Still Cameras:
Demosaicking and De-ringing**

Wang Xiaomeng

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University
in fulfilment of the requirement for the degree of
Master of Engineering

2005

TA
1637
W246
2005

Acknowledgements

Throughout my studies as a Master of Engineering student, I have received help and support from many people. First of all, I would like to thank my supervisors, Associate Professor Xue Ping (School of Electrical & Electronic Engineering, Nanyang Technological University) and Dr. Lin Weisi (Institute for Infocomm Research) for their invaluable advice, encouragement and guidance in the development of the project and throughout my studies.

I'm grateful to the staff from Information Systems Research Lab, Media Technology Lab, School of EEE, NTU for their kind assistance and support.

Finally, I'd like to thank all of my friends, who shared with me the happy time in NTU, and my parents, who are always encouraging me to get through the difficulties with their great support.

Table of Contents

List of Figures	iv
List of Tables	vi
Summary	vii
1. Introduction	1
1.1 Motivation	1
1.2 Objectives	2
1.3 Contributions	3
1.4 Organization of the thesis	3
2. Color Fundamentals & Perceptual Characteristics	5
2.1 Retinal receptors of the human eye	5
2.2 Color matching	7
2.3 The RGB color model	7
3. Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras	10
3.1 Color recording in digital still cameras	11
3.2 Color filter array (CFA)	12
3.3 Demosaicking methods in Bayer array	13
3.3.1 Bilinear interpolation	15
3.3.2 Smooth hue transition interpolation	16
3.3.3 Gradient based interpolation	17
3.3.4 Adaptive color plane interpolation	19

3.3.5	Combination of smooth hue transition and edge-directed interpolation	21
3.3.6	Alternating projections color plane interpolation	24
3.4	Simulation of various interpolation methods	27
4.	Color Demosaicking with Improved Edge-Direction Detection	34
4.1	Narrow-edge detection	34
4.2	A measure of local inter-channel correlation	39
4.3	G channel reconstruction	39
4.4.	R and B channel reconstruction	40
4.5	Refinement	41
4.6	Experimental results	42
4.6.1	Performance measure	42
4.6.2	Experimental results	43
5.	Ringling Artifacts Reduction in JPEG Compressed Digital Images	59
5.1	Image compression in digital still cameras	59
5.1.1	JPEG still picture compression standard	60
5.1.2	Compression artifacts	64
5.2	Proposed spatial adaptive de-ringing algorithm	65
5.2.1	Pixel classification	66
5.2.2	De-ringing	68
5.3	Experimental results	69
6.	Conclusion and Future Recommendation	75
	Bibliography	77

List of Figures

Figure 2.1: Color sensitivity of the human eye6

Figure 2.2: The scotopic and photopic sensitivities of the Human eye6

Figure 2.3: The color matching functions of the CIE RGB standard8

Figure 2.4: The RGB Color Cube9

Figure 3.1: An image processing pipeline for a digital photography system10

Figure 3.2: Bayer CFA pattern13

Figure 3.3: Illustration of mosaicking artifacts14

Figure 3.4: Bayer Pattern15

Figure 3.5: Pixel P_5 is the pixel that will be interpolated21

Figure 3.6: Simulation results of lighthouse image with different demosaicking
methods28

Figure 3.7: Simulation results of sailboat image with different demosaicking .
methods30

Figure 4.1: Bayer Color Filter Array35

Figure 4.2: The behaviors of first-order and second-order derivatives as they
encounter different objects38

Figure 4.3: Interpolation results of *Ray* image using different edge detectors44

Figure 4.4: Images used in experiments46

Figure 4.5: Average PSNR difference between the proposed method and other relative
methods in Table 4.3.....50

Figure 4.6: Visual comparison of *Image 1* (384×256 lighthouse image)51

Figure 4.7: Visual comparison of *Image 4* (768×512 lighthouse image)53

Figure 4.8: Visual comparison of *Image 7*55

Figure 4.9: Visual comparison of *Image 13*57

Figure 5.1: DCT-Based encoder processing steps61

Figure 5.2: DCT-Based decoder processing steps61

Figure 5.3: The zig-zag pattern of Entropy Encoding64

Figure 5.4: Luminance (Y) values in a 3×3 window69

Figure 5.5: Detected ringing artifacts around edges70

Figure 5.6: The luminance of JPEG compressed *cameraman* image: before and after
de-ringing post processing72

Figure 5.7: The luminance of JPEG compressed *plane* image: before and after
de-ringing post processing73

Figure 5.8: The luminance of JPEG compressed *sailboat* image: before and after
de-ringing post processing74

List of Tables

Table 4.1: A simplification of a gray-level image strip37

Table 4.2: PSNR comparison for interpolation with different edge detectors43

Table 4.3: PSNR comparison of different demosaicking methods49

Table 5.1: Quantization table used in the experiment70

Table 5.2: PSNR comparison of luminance (Y) between the JPEG decoded and post
processed images71

Summary

The digital still color camera has become a popular consumer product for capturing natural images. In a typical digital camera, the image is captured, processed, and then stored on the camera. To achieve cost reduction and compact design, digital still color cameras capture the visible spectrum by using a color filter array (CFA) overlaid on a single-sensor such that each pixel samples only one color channel (red, green or blue). The values of the other color channels must be interpolated using neighboring samples. This color plane interpolation is known as *demosaicking*. In order to maximize the efficiency of storage in a digital camera, images are usually compressed using a standard image compression algorithm, such as JPEG (Joint Photographic Experts Group).

In general, both the demosaicking process and the compression process produce their own artifacts that degrade the visual quality of the final image. Therefore, to improve the visual quality of images taken by the digital camera, the needs for good demosaicking techniques and post-processing schemes for JPEG compressed images are essential.

In the first part of this study, demosaicking methods for the single-sensor digital camera based upon Bayer color filter array (CFA) are investigated. Since accurate edge restoration is crucial to human perception, a new improved edge-adaptive demosaicking

algorithm emphasizing on more accurate edge direction detection by making full use of the intra- and inter-channel correlations is proposed.

In the second part of this study, a new spatial adaptive post-processing algorithm focusing on ringing artifacts reduction by exploiting the local statistics of the JPEG compressed image is proposed. This de-ringing scheme can reduce ringing artifacts around edges and prevent over smoothing texture regions in JPEG compressed images.

The visual quality of the images taken by the digital still camera can be improved by proposed demosaicking method for more accurate image reconstruction. The proposed de-ringing post processing scheme reduces P E G compression artifacts also.

Chapter 1

Introduction

1.1 Motivation

In the past twenty years, one major technological breakthrough in consumer electronics is to convert conventional analog information (represented by a gradually fluctuating wave) into digital information (represented by 1s and 0s, or bits). This fundamental shift in technology totally changed how we handle visual and audio information.

The digital still camera (DSC) is one of the most remarkable instances of this shift because the working principle of the DSC is so different from that of the conventional camera. Conventional cameras depend entirely on chemical and mechanical processes, while digital cameras record images in an entirely electronic form. Digital cameras are becoming ever more popular and replacing conventional film-based cameras in many applications.

In a typical digital camera, the image is captured, processed, and then stored on the camera [58]. To reduce the cost and size, most digital cameras employ a single sensor, either a CCD (Charge Coupled Devise) or a CMOS (Coupled Metal Oxide Semiconductor), which is sensitive only to the intensity of the illumination, and not the wavelength. Thus to gain color capacities, the sensor is overlaid with a color filter array (CFA), such that each pixel

Chapter1 Introduction

has its own color. The image captured by the sensor overlaid with CFA is therefore a *mosaiced* image, where each pixel holds the value for either red, green or blue. This image is often called the *raw data* image. The process of interpolating the *raw data* into a full color image is called *demosaicking*. After the *raw data* image is demosaicked, it is usually processed through source correction (color conversion operation and Gamma correction). Then the resulting image is stored in the camera. In order to maximize the efficiency of storage, the image is compressed using a standard image compression algorithm, usually JPEG (Joint Photographic Experts Group) [29, 30].

In general, both the demosaicking process and the compression process produce their own artifacts that degrade the visual quality of the final image. Therefore, to improve the visual quality of images taken by the digital camera, the needs for good demosaicking techniques and post-processing schemes for JPEG compressed images are essential.

1.2 Objectives

The purpose of this research is to improve the visual quality of images acquired from a digital camera by investigating and developing the image processing techniques in a digital photography system.

The main objective of this thesis is to compare and improve the demosaicking methods for more accurate image reconstruction, develop the post-processing scheme on JPEG

Chapter1 Introduction

compressed images to reduce compression artifacts and thus improve the visual quality of the final images taken by digital still cameras.

1.3 Contributions

The contributions of this thesis are summarized as follows:

- Some of the well known demosaicking methods are investigated and compared.
- A more accurate color edge direction detector for Bayer CFA in digital still cameras has been derived for narrow edges by making full use of inter-channel correlation. This helps to preserve fine edge details in the reconstructed images.
- An improved edge-adaptive color demosaicking method for Bayer pattern images from single-sensor digital cameras is proposed. The effort has been directed in better edge direction detection, emphasis on G channel restoration and overall refinement step.
- A spatial adaptive post-processing algorithm focusing on ringing artifacts reduction by exploiting the local statistics of the JPEG compressed image is developed.

1.4 Organization of the thesis

The rest of the thesis is organized as follows: Chapter 2 gives a review of color fundamentals and perceptual characteristics, which forms a basis for the rest of the thesis. Chapter 3 gives an introduction on color capturing techniques in digital still cameras and investigates some well known demosaicking methods. A new improved edge-adaptive color demosaicking method for Bayer CFA in digital still cameras is proposed and the

Chapter1 Introduction

experimental results are presented in chapter 4. Chapter 5 introduces the background knowledge on JPEG still picture compression standard used in digital still cameras. A new spatial adaptive post-processing algorithm focusing on ringing artifacts reduction for JPEG compressed images is developed also. Chapter 6 gives some conclusion remarks and future recommendation.

Chapter 2

Color Fundamentals & Perceptual Characteristics

2.1 Retinal receptors of the human eye

The innermost membrane of the eye is the retina, which lines the inside of the wall's entire posterior portion. When the eye is properly focused, light from an object outside the eye is imaged on the retina. Pattern vision is afforded by the distribution of discrete light receptors over the surface of the retina. There are two classes of receptors: *cones* and *rods* [1].

Cones are the sensors in the eye responsible for color vision, which can be divided into three principal sensing categories, namely the L, M, S based on their independent sensitivities to different wavelengths of light (Figure 2.1), corresponding roughly to red, green, and blue. Due to the absorption characteristics of the human eye, colors are seen as variable combination of the so called *primary colors* red (*R*), green (*G*), and blue (*B*). Cone vision is called *photopic* or bright-light vision.

Rods serve to provide monochromatic vision under low illumination levels, they are not involved in color vision. Objects that appear brightly colored in daylight when seen by moonlight appear as colorless forms because only the rods are stimulated. This phenomenon is known as *scotopic* or dim-light vision [1]. The spectral sensitivity of the cones and rods is shown in Figure 2.2.

Chapter 2 Color Fundamentals & Perceptual Characteristics

Figure 2.1: Color sensitivity of the human eye

Figure 2.2: The scotopic and photopic sensitivities of the Human eye.

Chapter 2 Color Fundamentals & Perceptual Characteristics

2.2 Color matching

Color vision is basically a function of three variables. As mentioned before, the cones can be separated into three classes, each class being sensitive to a different spectral distribution of radiation. Although the rods provide a fourth spectrally different receptor, at levels of illumination that are high enough for color vision to be operating properly, there is evidence that the outputs of the rods are rendered ineffective [1]. Hence, it is expected that the evaluation of perceived color from spectral power data should require the use of three different spectral weighting functions. Because of the phenomenon of trichromacy, any color stimulus can be matched by a mixture of three primary stimuli.

2.3 The RGB color model

A color model is a 3D unique representation of a color. There are different color models and the use of one over the other is problem oriented. For instance, the color model RGB is used in cameras.

In the RGB model, each color channel appears in its primary spectral components of red, green, and blue. The human eye perceives wavelengths of 400-500 nm (nanometers) as blue, 500-600 nm as green, and 600-700 nm as red. Combining light of these three wavelengths in varying intensities can reproduce all colors encountered in nature. Mixing 100% of all three colors will produce white light. Reducing each component to 0% will result in the absence of light, or black. It was observed that it was possible to map the

Chapter 2 Color Fundamentals & Perceptual Characteristics

visible color spectrum using the $r(\lambda)$, $g(\lambda)$, $b(\lambda)$ weighting functions, known as *color-matching functions*, shown in Figure 2.3. These colors are called *additive* colors because they can be added to produce different colors; meaning, if one power unit of one wavelength λ_1 is matched by

$$r_1 \text{ of R} + g_1 \text{ of G} + b_1 \text{ of B} \tag{2.1}$$

and one power unit of light of wavelength λ_2 is matched by

$$r_2 \text{ of R} + g_2 \text{ of G} + b_2 \text{ of B} \tag{2.2}$$

then the additive mixture of the two lights, λ_1 and λ_2 , is matched by

$$(r_1 + r_2) \text{ of R} + (g_1 + g_2) \text{ of G} + (b_1 + b_2) \text{ of B} \tag{2.3}$$

This means that the color-matching functions of Figure 2.3 can be used as weighting functions to determine the amounts of R, G, B needed to match any color.

Figure 2.3: The color matching functions of the CIE RGB standard

Chapter 2 Color Fundamentals & Perceptual Characteristics

In the R, G, B color model described above, the amounts of R, G and B are referred to as *tristimulus values*. In general, a display system has stimuli values that can vary from 0 to 255. This is represented as an RGB color cube as shown in Figure 2.4. The RGB Color model that is used in color displays, use tristimulus values in the range of [0,255].

Figure 2.4: The RGB Color Cube

Chapter 3

Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

In a typical digital camera, the image is captured, processed, and then stored on the camera. In order to maximize the efficiency of storage, the camera can use a standard image compression algorithm, usually JPEG, on the image data. Figure 3.1 shows the key image processing steps implemented in the digital still cameras and host computers for a digital photography system [58].

Figure 3.1: An image processing pipeline for a digital photography system.

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

3.1 Color recording in digital still cameras

Digital still cameras use electronic sensors to acquire the spatial variations in light intensity and then use image processing algorithms to reconstruct a color picture from the data provided by sensors. In order to get a full color image, most sensors use filtering to acquire the light intensity in its three primary colors: red, green and blue. Once all three color components have been recorded, they can be added together to create the full spectrum of colors.

There are several ways of recording the three primary colors in a digital camera [59]. The highest quality cameras use three separate sensors, each with a different filter over it. A beam splitter is placed in the camera to separate the light from the camera lens into RGB component images that are focused onto three separate monochrome sensors [2]. The advantage of this method is that the camera records each of the three color channels at each pixel location. Unfortunately, cameras that use this method tend to be bulky and expensive.

A second method is to rotate a series of red, blue and green filters in front of a single monochrome sensor. The sensor records three separate images in rapid succession. This method also provides information on all three colors at each pixel location; but since the three images aren't taken at precisely the same moment, both the camera and the target of the photo must remain stationary for all three readings. These cameras are used for studio photograph of inanimate scenes, but are not practical for candid photography or handheld cameras.

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

A more economical and practical way to record the three primary colors from a single image is to capture the image using a single sensor photosites on top of which RGB color filters have been fabricated. The use of interleaved color filters creates three distinct sensor classes, and each photosite is sensitive to only one spectral band. This is called a mosaiced design — a mosaic of colored filters (i.e., a color filter array, or CFA). When a mosaiced design is used, the captured sampled image must be interpolated to produce a full-color image. This color plane interpolation is called *demosaicking*.

3.2 Color filter array (CFA)

To reconstruct a color image, we need to sample luminance and chrominance information. Since the Human Visual System (HVS) is more sensitive to the luminance information than the chrominance information (rod cells are much more sensitive to light than cones and are much more numerous), the luminance channel may be sampled at a rate much higher than the chrominance channels. And it has been shown that the spectral sensitivity of the HVS to luminance is similar to the spectral power distribution of the green channel. Hence, green sensory data are treated as the luminance channel, and red and blue sensory data are treated as the chrominance channels. According to these conditions, a number of color filter arrays have been invented.

The most commonly used spatial arrangement of color filters in a digital camera is the Bayer CFA, named after its inventor [3] (see Figure 3.2). This CFA has two times more green filters than red and blue filters. It is composed of alternate lines of two primary colors: red/green on odd lines, and blue/green on even lines.

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

Figure 3.2: Bayer CFA pattern

Filters that are selectively transparent in the green region of the spectrum are preferably employed in producing luminance elements, and filters that are selectively transparent in the red and blue spectral regions, respectively, are preferably employed in producing chrominance elements. (The term “luminance” here is used in a broad sense to refer to the color vector that is the major contributor of luminance information, while the term “chrominance” refers to the color vectors other than the luminance color vectors) [3].

3.3 Demosaicking methods in Bayer array

Demosaicking refers to the process that allows recreating a three-primary color per pixel image from a one-primary color per pixel image.

Figure 3.3 shows the result obtained from demosaicking using linear interpolation. The original image is shown in Figure 3.3a, a region of interest of which is extracted in Figure 3.3b. Linear interpolation gives rise to artifacts in the image as shown in Figure 3.3c. These images are included here to give the reader an idea of the problem at hand.

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

Figure 3.3: Illustration of demosaicking artifacts

- (a) original reference image
- (b) a small portion from the original image
- (c) result of linear interpolation

There are a number of demosaicking methods available, the simplest being linear interpolation, as shown in Figure 3.3, such a simple method fails to restore high-frequency components and thus introduces strongly visible color aliasing artifacts, especially around object edges. Then more complicated methods [4-27] are proposed to attempt to limit hue transitions or preserve high-frequency details. In this section, we will introduce some existing demosaicking methods.

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

3.3.1 Bilinear interpolation

Consider the Bayer pattern array shown in Figure 3.4.

B ₂₂	G ₂₃	B ₂₄	G ₂₅	B ₂₆
G ₃₂	R ₃₃	G ₃₄	R ₃₅	G ₃₆
B ₄₂	G ₄₃	B ₄₄	G ₄₅	B ₄₆
G ₅₂	R ₅₃	G ₅₄	R ₅₅	G ₅₆
B ₆₂	G ₆₃	B ₆₄	G ₆₅	B ₆₆

Figure 3.4: Bayer Pattern

At a blue center, we need to estimate the green and red components. Consider pixel 44 at which B₄₄ is measured; we need to determine G₄₄. Given G₃₄, G₄₃, G₄₅, G₅₄, the estimate for G₄₄ is given by

$$G_{44} = \frac{G_{34} + G_{43} + G_{54} + G_{45}}{4} \quad (3.1)$$

To determine R₄₄, given R₃₃, R₃₅, R₅₃, R₅₅, the estimate for R₄₄ is given by

$$R_{44} = \frac{R_{33} + R_{35} + R_{53} + R_{55}}{4} \quad (3.2)$$

At a red center, we would estimate the blue and green accordingly. Performing this process at each pixel, we can obtain three-primary color planes for the scene that would give us one possible demosaicked form of the scene.

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

Although this method is very simple, as can be expected, such a simple method fails to restore high-frequency components and thus introduces visible color aliasing artifacts, especially around object edges.

3.3.2 Smooth hue transition interpolation

More sophisticated demosaicking methods attempt to limit hue transitions. This *smooth hue transition* algorithm is proposed by Cok [4] and is one of the first few methods used in commercial camera systems.

This algorithm is based on the assumption that the color ratio within a given object is constant, and there are no sudden jumps in hue between neighboring pixel locations. The term *hue* here is defined by R/G, B/G. By interpolating the hue value and deriving the interpolated chrominance values (blue and red) from the interpolated hue values, hues are allowed to change only gradually, thereby reducing the appearance of color fringes in areas of image details. In this method, G channel is interpolated with bilinear interpolation, and then R and B channels are interpolated by color ratios (R/G, B/G) to avoid false color.

Still referring to Figure 3.4, the green channel is interpolated first using bilinear interpolation.

$$G_{44} = \frac{G_{34} + G_{43} + G_{54} + G_{45}}{4} \quad (3.3)$$

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

Then the hue (R/G, B/G) is interpolated

$$R_{44} = G_{44} \cdot \frac{\left(\frac{R_{33}}{G_{33}} + \frac{R_{35}}{G_{35}} + \frac{R_{53}}{G_{53}} + \frac{R_{55}}{G_{55}} \right)}{4} \quad (3.4)$$

$$B_{33} = G_{33} \cdot \frac{\left(\frac{B_{22}}{G_{22}} + \frac{B_{24}}{G_{24}} + \frac{B_{42}}{G_{42}} + \frac{B_{44}}{G_{44}} \right)}{4} \quad (3.5)$$

This approach gives better results than pure bilinear interpolation and does reduce hue transitions over the edges since it interpolates not the colors, but the hue of the colors, which reduces abrupt hue jumps, hence producing fewer perceptual artifacts. However, artifacts are still visible around sharp transition or edge regions where the assumption of constant hue does not hold.

3.3.3 Gradient based interpolation

In order to improve the quality in reconstructing edge regions, some *edge-sensing* algorithms have been developed [5, 7, and 10]. The method proposed by Laroche and Prescott [5] is in use in the Kodak DCS 200 Digital Camera System. It employs a three-step process, the first one being the interpolation of the luminance channel (green), the second and third being bilinear interpolation of the chrominance channels (red and blue). This method takes advantage of the fact that the human eye is most sensitive to luminance (green) changes, and depending upon the position of an edge in the green channel, the interpolation is performed accordingly.

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

Referring back to Figure 3.4, if we need to estimate G_{44} , let

$$\Delta H = \left| \frac{(B_{42} + B_{46})}{2} - B_{44} \right| \quad (3.6)$$

$$\Delta V = \left| \frac{(B_{24} + B_{64})}{2} - B_{44} \right| \quad (3.7)$$

ΔH And ΔV are estimates to the horizontal and vertical second-derivatives respectively, in blue. Using these second derivatives as classifiers, we come up with the following estimates for the missing green pixel value.

$$G_{44} = \begin{cases} \frac{(G_{43} + G_{45})}{2} & \text{if } (\Delta H < \Delta V) \\ \frac{(G_{34} + G_{54})}{2} & \text{if } (\Delta H > \Delta V) \\ \frac{(G_{43} + G_{45} + G_{34} + G_{54})}{4} & \text{if } (\Delta H = \Delta V) \end{cases} \quad (3.8)$$

Similarly, for estimating G_{33} , let

$$\Delta H = \left| \frac{(R_{31} + R_{35})}{2} - R_{33} \right| \quad (3.9)$$

$$\Delta V = \left| \frac{(R_{13} + R_{53})}{2} - R_{33} \right| \quad (3.10)$$

$$G_{33} = \begin{cases} \frac{(G_{32} + G_{34})}{2} & \text{if } (\Delta H < \Delta V) \\ \frac{(G_{23} + G_{43})}{2} & \text{if } (\Delta H > \Delta V) \\ \frac{(G_{32} + G_{34} + G_{23} + G_{43})}{4} & \text{if } (\Delta H = \Delta V) \end{cases} \quad (3.11)$$

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

Once the luminance is determined, the chrominance values are interpolated from the color differences between the chrominance (red and blue) and luminance (green) signals.

This is given as follows:

$$R_{34} = G_{34} + \left(\frac{(R_{33} - G_{33}) + (R_{35} - G_{35})}{2} \right) \quad (3.12)$$

$$R_{43} = G_{43} + \left(\frac{(R_{33} - G_{33}) + (R_{53} - G_{53})}{2} \right) \quad (3.13)$$

$$R_{44} = G_{44} + \left(\frac{(R_{33} - G_{33}) + (R_{53} - G_{53}) + (R_{35} - G_{35}) + (R_{55} - G_{55})}{4} \right) \quad (3.14)$$

The interpolation of B channel is performed in a similar way. Laroche-Prescott's algorithm, first detects the second derivatives of R or B channel in a local neighborhood, and then makes a choice of neighboring pixels to interpolate the missing color values in such a way that the interpolation is carried along the edge direction, rather than across edges.

3.3.4 Adaptive color plane interpolation

This method [10], proposed by Hamilton and Adams, is a modification of the above method [5]. In addition to using second derivatives as the estimate, the missing green values are estimated by a combination of second derivatives in R, B channel and the green channel gradient estimate. The inter-channel correlation is also used as a correction term in the interpolation.

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

This process also has three steps. The first step interpolates the luminance (green) channel and the second and third steps interpolate the chrominance (red and blue) channels. Referring back to Figure 3.4, define the following classifiers,

$$\Delta H = |-B_{42} + 2B_{44} - B_{46}| + |G_{43} - G_{45}| \quad (3.15)$$

$$\Delta V = |-B_{24} + 2B_{44} - B_{64}| + |G_{34} - G_{54}| \quad (3.16)$$

These classifiers are composed of second derivative terms for chromaticity data and gradients for the luminance data. If we need to estimate the green value at the center B_{44} ,

$$G_{44} = \begin{cases} \frac{(G_{43} + G_{45})}{2} + \frac{(-B_{42} + 2B_{44} - B_{46})}{4} & \text{if } (\Delta H < \Delta V) \\ \frac{(G_{34} + G_{54})}{2} + \frac{(-B_{24} + 2B_{44} - B_{64})}{4} & \text{if } (\Delta H > \Delta V) \\ \frac{(G_{43} + G_{45} + G_{34} + G_{54})}{4} + \frac{(-B_{42} - B_{24} + 4B_{44} - B_{46} - B_{64})}{8} & \text{if } (\Delta H = \Delta V) \end{cases} \quad (3.17)$$

This comprises the first step of the interpolation algorithm. The second and third steps perform the interpolation of chromaticity channels.

$$R_{34} = \frac{(R_{33} + R_{35})}{2} + \left(\frac{-G_{33} + 2G_{34} - G_{35}}{2} \right) \quad (3.18)$$

$$R_{43} = \frac{(R_{33} + R_{53})}{2} + \left(\frac{-G_{33} + 2G_{43} - G_{53}}{2} \right) \quad (3.19)$$

In order to calculate R_{44} , define α, β which estimate the gradient in the positive and negative diagonal directions respectively.

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

$$\alpha = |-G_{35} + 2G_{44} - G_{53}| + |R_{35} - R_{53}| \quad (3.20)$$

$$\beta = |-G_{33} + 2G_{44} - G_{55}| + |R_{33} - R_{55}| \quad (3.21)$$

Then R_{44} is calculated as the following equations,

$$R_{44} = \begin{cases} \frac{(R_{35} + R_{53})}{2} + \frac{(-G_{35} + 2G_{44} - G_{53})}{2} & \text{if } (\alpha < \beta) \\ \frac{(R_{33} + R_{55})}{2} + \frac{(-G_{33} + 2G_{44} - G_{55})}{2} & \text{if } (\alpha > \beta) \\ \frac{(R_{33} + R_{35} + R_{53} + R_{55})}{4} + \frac{(-G_{33} - G_{35} + 4G_{44} - G_{53} - G_{55})}{4} & \text{if } (\alpha = \beta) \end{cases} \quad (3.22)$$

At a red center, we would estimate the blue and green accordingly.

3.3.5 Combination of smooth hue transition and edge-directed interpolation

Another method proposed by Kimmel [12] combines the smooth hue transition and edge-directed interpolation in an iterative scheme based on the assumption that the colors are smooth within a given object and go through a sudden jump at the boundaries. The method use the gradient magnitude as an edge indicator, the direction of the gradient magnitude can approximate the edge direction.

P1	P2	P3
P4	P5	P6
P7	P8	P9

Fig 3.5: Pixel P_5 is the pixel that will be interpolated.

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

The directional derivatives are estimated at each point based on its eight nearest neighboring pixels. From Figure 3.5, define finite difference approximation for the directional derivatives at pixel P_5 , in the x , y , x -diagonal and y -diagonal direction as follows:

$$\begin{aligned} D_x(P_5) &= \frac{P_4 - P_5}{2} & D_y(P_5) &= \frac{P_2 - P_8}{2} \\ D_{xd}(P_5) &= \frac{P_3 - P_7}{2\sqrt{2}} & D_{yd}(P_5) &= \frac{P_1 - P_9}{2\sqrt{2}} \end{aligned} \quad (3.23)$$

Notice that differences are always from the same color plane. If pixel P_5 is a green pixel then so are P_1, P_3, P_7 and P_9 . To obtain the gradient in the diagonal direction at a green pixel, it defines the diagonal gradients as:

$$D_{xd}(P_5) = \max\left\{\left|\frac{P_3 - P_5}{\sqrt{2}}\right|, \left|\frac{P_7 - P_5}{\sqrt{2}}\right|\right\} \quad D_{yd}(P_5) = \max\left\{\left|\frac{P_1 - P_5}{\sqrt{2}}\right|, \left|\frac{P_9 - P_5}{\sqrt{2}}\right|\right\} \quad (3.24)$$

With the above gradients, Kimmel defines edge indicator function e_i as:

$$e_i = \frac{1}{\sqrt{1 + D(P_5)^2 + D(P_i)^2}} \quad (3.25)$$

Where D is the difference in the direction of P_i . Here are two examples,

$$e_6 = \frac{1}{\sqrt{1 + D_x(P_5)^2 + D_x(P_6)^2}} \quad (3.26)$$

$$e_3 = \frac{1}{\sqrt{1 + D_{xd}(P_5)^2 + D_{xd}(P_3)^2}} \quad (3.27)$$

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

Based on the edge indicators as weighting factors for the interpolation, the interpolation is performed as follows:

- *Init*: First interpolate the green value at pixel P_5 ,

$$G_5 = \frac{e_2 G_2 + e_4 G_4 + e_6 G_6 + e_8 G_8}{e_2 + e_4 + e_6 + e_8} \quad (3.28)$$

For the blues, first interpolate the missing blues at the red locations

$$B_5 = \frac{e_1 \frac{B_1}{G_1} + e_3 \frac{B_3}{G_3} + e_7 \frac{B_7}{G_7} + e_9 \frac{B_9}{G_9}}{e_1 + e_3 + e_7 + e_9} \quad (3.29)$$

Second, interpolate the missing blues at the green locations.

$$B_5 = \frac{e_2 \frac{B_1}{G_1} + e_4 \frac{B_3}{G_3} + e_6 \frac{B_7}{G_7} + e_8 \frac{B_9}{G_9}}{e_2 + e_4 + e_6 + e_8} \quad (3.30)$$

Then interpolate missing red pixels similar to the blue pixels.

- *Repeat for three times*:
- Correct the green values to fit the ratio rule

$$G_5^B = B_5 \frac{\sum e_i \frac{G_i}{B_i}}{\sum e_i}, \text{ with } i \neq 5 \quad (3.31)$$

$$G_5^R = B_5 \frac{\sum e_i \frac{G_i}{R_i}}{\sum e_i}, \text{ with } i \neq 5 \quad (3.32)$$

and average between the Blue and Red interpolation results

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

$$G_s = \frac{G_s^B + G_s^R}{2} \quad (3.33)$$

- Correct the blue and red values via the ratio rule weighted by the edge indicator

$$B_s = G_s \frac{\sum e_i \frac{B_i}{G_i}}{\sum e_i}, \text{ with } i \neq 5 \quad (3.34)$$

$$R_s = G_s \frac{\sum e_i \frac{R_i}{G_i}}{\sum e_i}, \text{ with } i \neq 5 \quad (3.35)$$

- End of loop.

This weighted sum algorithm takes advantage of the assumption that the colors are smooth within a given object and go through a sudden jump at the boundaries. The method finds the likelihood of an edge in a certain direction, and performs the interpolation based on the edge likelihoods. The pixels along the direction of a more likely edge are given higher weights in the interpolating formula, and vice-versa.

3.3.6 Alternating projections color plane interpolation

This method [17], proposed by B. K. Gunturk, Y. Altunbasak and R. M. Mersereau, has been reported to be able to achieve superior demosaicking results as compared with other existing methods.

The alternating projections interpolation method exploits the high spectral correlation between different color planes for estimating the missing color values by alternately

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

projecting their estimates onto two constraint sets. The first constraint set comes from the observed color samples, which means the color values in the demosaicked color planes must be consistent with the original CFA samples. The second constraint set, an exploitation of the high image spectral correlation, is defined on the red and blue channels that force their high-frequency components to be similar to the high-frequency components of the green channel. To meet the second constraint set, each color plane is decomposed using a low-pass and a high-pass analysis filters into four subbands: 1) LL: an approximation subband whose rows and columns are low-pass filtered; 2) LH: a vertical subband whose rows are low-pass filtered and columns are high-pass filtered; 3) HL: a horizontal subband whose rows are high-pass filtered and columns are low-pass filtered; 4) HH: a diagonal subband whose rows and columns are high-pass filtered.

Then the algorithm works as follows:

- 1). Initial interpolation: This step uses an edge adaptive demosaicking method [10] to interpolate the green channel and bilinear method to interpolate the red and blue channels to obtain initial estimates.
- 2). Update the green channel:
 - (a) Use the observed samples of the red channel to form a downsampled version of the red channel. Use the interpolated green samples at the corresponding red locations to form a downsampled version of the green channel.

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

- (b) Decompose the these red and green downsampled channels into four subbands (LL, LH, HL, and HH), given by

$$\begin{aligned} S_{LL}(n_1, n_2) &= h_0(n_1) * [h_0(n_2) * S(n_1, n_2)] \\ S_{LH}(n_1, n_2) &= h_0(n_1) * [h_1(n_2) * S(n_1, n_2)] \\ S_{HL}(n_1, n_2) &= h_1(n_1) * [h_0(n_2) * S(n_1, n_2)] \\ S_{HH}(n_1, n_2) &= h_1(n_1) * [h_1(n_2) * S(n_1, n_2)] \end{aligned} \quad (3.36)$$

where (n_1, n_2) denotes the pixel coordinates, $h_0 = [1 \ 2 \ 1]/4$ and $h_1 = [1 \ -2 \ 1]/4$ are the low-pass and high-pass analysis filters respectively, and S designates the two downsampled color channels, R_r and G_r .

- (c) Replace the high-frequency (LH, HL, and HH) subbands of the G_r plane with that of the R_r plane.
- (d) Reconstruct the G_r plane by using Eq. (3.37), where $g_0 = [-1 \ 2 \ 6 \ 2 \ -1]/8$ and $g_1 = [1 \ 2 \ -6 \ 2 \ 1]/8$ are the low-pass and high-pass synthesis filters respectively, and insert the reconstructed green values into their corresponding locations in the initial green channel estimate.

$$\begin{aligned} G_r(n_1, n_2) &= g_0(n_1) * [g_0(n_2) * G_{r_{LL}}(n_1, n_2)] + g_0(n_1) * [g_1(n_2) * R_{r_{LH}}(n_1, n_2)] \\ &\quad + g_1(n_1) * [g_0(n_2) * R_{r_{HL}}(n_1, n_2)] + g_1(n_1) * [g_1(n_2) * R_{r_{HH}}(n_1, n_2)] \end{aligned} \quad (3.37)$$

- (e) The updating of the estimated green values at blue pixels is similarly performed using the downsampled blue plane.

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

- 3). “Detail” projection: Decompose all three color channels into four subbands, and replace the detail (LH, HL, and HH) subbands of the red and blue channels with that of the green channel and then reconstruct these channels.
- 4). “Observation” projection: Compare the samples of the reconstructed red and blue channels with the original samples. Insert the original samples into the reconstructed channels at their corresponding pixel locations to preserve the observed values.
- 5). Iteration: Repeat the detail projection step and the observation projection step for a preset number of times.

3.4 Simulation of various interpolation methods

As shown in Figure 3.6 and Figure 3.7, the above interpolation methods are simulated by using *lighthouse* and *sailboat* images. These original full color images were first sampled with Bayer CFA pattern to produce mosaic images, and then these mosaic images were used as input raw data for simulating different demosaicking methods.

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

(a) Original *lighthouse* image

(b) Magnified details of the original image

(c) Bilinear interpolation

(d) Cok's method

Figure 3.6: Simulation results of *lighthouse* image with different demosaicking methods.

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

(e) Laroche and Prescott's method

(f) Hamilton and Adams' method

(g) Kimmel's method

(h) Gunturk et al.'s method

Figure 3.6: (Continued) Simulation results of *lighthouse* image with different demosaicking methods.

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

(a) Original *sailboat* image

(b) Magnified details of the original image

(c) Bilinear interpolation

(d) Cok's method

Figure 3.7: Simulation results of *sailboat* image with different demosaicking methods.

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

(e) Laroche and Prescott's method

(f) Hamilton and Adams' method

(g) Kimmel's method

(h) Gunturk et al.'s method

Figure 3.7: (Continued) Simulation results of *sailboat* image with different demosaicking methods.

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

By comparing the simulation results of the above interpolation methods, we can observe that there are two main types of artifacts in the demosaicked images: zipper effects and false colors. Zipper effects are characterized by the lines of dots around sharp edges in regular intervals (one dot every two pixels). And the false colors are noticeable color errors as compared to the original non-mosaicked image.

As shown in Figure 3.6 and Figure 3.7, bilinear method, Cok's method and Gunturk et al.'s method conduct visible zipper effects around sharp edges, since these methods do not exploit the edge information effectively and perform the interpolation by using the cross-edge neighboring pixels improperly. It means that inadequate use of spatial correlation often results in zipper effects around the edge regions.

Concerning the false colors, the bilinear method exhibits the strongest visible false color artifacts as expected. Cok's method achieves better performance than pure bilinear interpolation by taking advantage of the spectral correlation (color ration rule). Laroche & Prescott's method, Hamilton & Adams' method and Kimmel's method combine both spatial correlation (edge direction estimation) and spectral correlation (color difference or color ratio rule) in the interpolation, therefore they produce fewer false color artifacts than bilinear and Cok's methods as shown in Figure 3.6–3.7 (e~g). Gunturk et al.'s method exploits spectral correlation by alternating projecting the estimates of the missing color values onto constraint sets based on original CFA samples and prior knowledge of

Chapter 3 Color Capturing Techniques and Demosaicking Methods in Digital Still Cameras

spectral correlation in an effective way. Experimental comparisons [17] have demonstrated that Gunturk et al.'s method can achieve superior performance than other existing method.

However, although such edge-sensing and alternating projections methods can achieve better quality in reconstructing edge regions than bilinear and smooth hue transition algorithms, they still fail to detect some fine image details which can cause visible false color artifacts as shown in the region around the window and the house siding in Figure 3.6 and the region around digits in Figure 3.7.

Chapter 4

Color Demosaicking with Improved Edge-Direction Detection

As we know human visual system (HVS) has increased sensitivity to luminance (green) changes when compared to chrominance (red and blue) changes, thus the reconstruction of green (G) channel information is very important to human perception. And since natural images are characterized by high correlation between their RGB color components, the interpolation of R and B channels takes advantage of the G channel information in most demosaicking methods except for the pure bilinear interpolation, the performance of the R and B channel reconstruction directly depends on the accuracy of the G channel reconstruction. Therefore green channel reconstruction is a crucial step in the entire demosaicking process. The key issue in G channel reconstruction is to detect occurrence of edges and avoid interpolation across edges. Taking these into account, we propose a new edge classifier to improve G channel reconstruction. The proposed edge classifier will make full use of the available chrominance and luminance information, as well as intra- and inter-channel correlation. The missing color values are interpolated along the edge direction, not across the edges, so that the color aliasing artifacts are reduced.

4.1 Narrow-edge detection

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

For *edge-sensing* demosaicking, interpolation is carried out along the detected edge direction. In the existing methods, edge information is derived from the same color plane [5, 7, 10, and 12].

G₁₁	R₁₂	G₁₃	R₁₄	G₁₅	R₁₆
B₂₁	G₂₂	B₂₃	G₂₄	B₂₅	G₂₆
G₃₁	R₃₂	G₃₃	R₃₄	G₃₅	R₃₆
B₄₁	G₄₂	B₄₃	G₄₄	B₄₅	G₄₆
G₅₁	R₅₂	G₅₃	R₅₄	G₅₅	R₅₆
B₆₁	G₆₂	B₆₃	G₆₄	B₆₅	G₆₆

Figure 4.1. Bayer Color Filter Array

For G channel estimation at a red center (R_{34}) in Figure 4.1, the G gradients (first derivatives) are calculated for horizontal and vertical directions to indicate occurrence of edges in the method [7]:

$$\Delta H_G = |G_{33} - G_{35}| \quad \text{and} \quad \Delta V_G = |G_{24} - G_{44}| \quad (4.1)$$

In the method [5], the G channel edge classifier is defined by detecting the second derivatives in R channel for the horizontal and vertical directions respectively.

$$\Delta H_R = |-R_{32} + 2R_{34} - R_{36}| \quad \text{and} \quad \Delta V_R = |-R_{14} + 2R_{34} - R_{54}| \quad (4.2)$$

As we know, the first derivative based detector (4.1) is able to find thick edges, and the second derivative based detector (4.2) can be used to find thinner edges. The *detection radius* in (4.1) and (4.2) is 3 pixels. As an improvement of [5, 7], the approach described in [10] derives the G channel edge detector by accumulating (4.1) and (4.2) for each

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

direction. Obviously, the smaller the *detection radius* of a second derivative based detector is, the finer the image details can be detected.

There are high correlations among RGB color channels in natural images, and the correlation here means that all the RGB components have similar derivatives. In a full color natural image, all the color channels are very likely to have the same edge content, for example, if the red channel value gets higher in one direction, then it is very probable that the green and blue channel values get higher in that direction too. Therefore we tend to make use of this kind of inter-channel correlation to detect fine details in a local region and then improve the accuracy of the edge detection.

In our method, a smaller *detection radius* can be achieved if the second derivatives are defined with inter-channel samples. We use the center chrominance (R or B) pixel and its neighboring G pixels to detect high-frequency information in the G value estimation at the available R or B pixel locations. For example, if we want to estimate G_{34} in Figure 4.1, the new edge detector is proposed as follows:

$$\Delta H_{GR} = |G_{33} + G_{35} - 2R_{34}| \quad \text{and} \quad \Delta V_{GR} = |G_{24} + G_{54} - 2R_{34}| \quad (4.3)$$

Notice that our ΔH_{GR} and ΔV_{GR} are defined from different color planes, by this way, the minimum *detection radius*—2 *pixels* can be achieved for ΔH_{GR} and ΔV_{GR} . Similar formulation can be made for a B centre.

In general, gradients and second derivatives are calculated in the same color plane. However, as we mentioned, in full color natural images, all the RGB color channels are very likely to have the same edge content, which means if the red channel value gets

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

higher in one direction, then it is very probable that the green and blue channel values get higher in that direction too. Therefore when only the mosaic images are available, the proposed narrow edge detector (4.3) from different color planes is an *approximate estimation* of gradient information in a 2 pixel width region, and this kind of *approximate estimation* does help to detect fine image details, as we'll see in the experimental results, especially in the high correlated image regions.

Image strip	5	4	3	2	1	0	0	0	6	0	0	0	6	0	6	0	2	3	2	0	0	0	0	7	7	7
ΔH_G	1	2	2	2	2	1	0	6	0	6	0	6	0	0	0	5	3	0	3	1	0	0	7	7	0	
ΔH_R		1	0	0	1	2	7	0	12	0	12	0	6	0	5	3	5	6	1	3	1	7	7	7	7	
ΔH_{GR}		0	0	0	0	1	0	6	12	6	0	6	12	12	12	7	1	2	1	1	0	0	7	7	0	

Table 4.1: A simplification of a gray-level image strip

Table 4.1 shows a simplification of a gray-level image strip for us to analyze how the first-order derivative and the second-order derivative behave as they encounter different objects. It also can be used to illustrate the behavior of ΔH_{GR} in the Bayer pattern because of the high correlation between RGB channels in natural images.

Figure 4.2 is a figure illustration for Table 4.1. First, we note that in Figure 4.2 the first order derivative (ΔH_G) is nonzero along the ramp, while the second-order derivative ($\Delta H_R, \Delta H_{GR}$) is almost zero, thus ΔH_G is used to detect thick edges (step change). At the following, the second-order derivatives ($\Delta H_R, \Delta H_{GR}$) have stronger responses to thin lines (sharp change) than the first order derivative (ΔH_G), therefore ΔH_R and ΔH_{GR} are used to detect thin lines. Especially, when encounter one pixel lines, i.e. fine details, ΔH_{GR} has lager values than the other two edge classifiers ΔH_G and ΔH_R , thus can achieve more accurate edge detection in fine detail image regions.

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

Figure 4.2: The behaviors of first-order and second-order derivatives as they encounter different objects.

4.2 A measure of local inter-channel correlation

In natural images, the correlation between G and R channels (or G and B channels) differs in different regions, and the color correlation is maximum when $R_{ij} = G_{ij} = B_{ij}$. Taking these into account we propose a measure of local inter-channel correlation for Bayer pattern. For simplicity, the local correlation measure is defined as the absolute difference between the average values of G and R, or G and B in a small region in Bayer pattern.

For example, at a red center (R_{34}) in Figure 4.1, the correlation measure between G and R, G and B in a 5×5 neighborhood is:

$$C_{GR} = |\overline{G} - \overline{R}| \quad \text{and} \quad C_{GB} = |\overline{G} - \overline{B}| \quad (4.4)$$

where $\overline{G}$, $\overline{R}$ and $\overline{B}$ are the average G, R and B values in the defined neighborhood. According to our definition, the smaller the C_{GR} or C_{GB} is, the higher the correlation between G and R or G and B is.

4.3 G channel reconstruction

Since there are various types of edges in real-world images, a single edge detector cannot cope with all situations. We therefore devise a combinative criterion for edge direction determination. For an R centre (R_{34}) in Figure 4.1, we define:

$$\Delta H = \Delta H_R + \Delta H_G + \begin{cases} \Delta H_{GR} & \text{if } C_{GR} \leq C_{GB} \\ \Delta H_{GB} & \text{otherwise} \end{cases} \quad (4.5)$$

$$\Delta V = \Delta V_R + \Delta V_G + \begin{cases} \Delta V_{GR} & \text{if } C_{GR} \leq C_{GB} \\ \Delta V_{GB} & \text{otherwise} \end{cases} \quad (4.6)$$

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

where

$$\Delta H_{GB} = \frac{1}{2} \left(\left| B_{23} + B_{25} - 2G_{24} \right| + \left| B_{43} + B_{45} - 2G_{44} \right| \right) \quad (4.7)$$

$$\Delta V_{GB} = \frac{1}{2} \left(\left| B_{23} + B_{43} - 2G_{33} \right| + \left| B_{25} + B_{45} - 2G_{35} \right| \right) \quad (4.8)$$

As our definition of the local inter-channel correlation measure, when $C_{GR} < C_{GB}$, the local correlation between G and R channels is higher than that between G and B channels, in that case ΔH_{GR} and ΔV_{GR} are used in the combinative edge direction detector, and vice versa.

The missing G component is interpolated as:

$$G_{34} = R_{34} + \begin{cases} \frac{e_{24}(G_{24} - R'_{24}) + e_{44}(G_{44} - R'_{44})}{e_{24} + e_{44}} & \text{if } \Delta H > \Delta V \\ \frac{e_{33}(G_{33} - R'_{33}) + e_{35}(G_{35} - R'_{35})}{e_{33} + e_{35}} & \text{if } \Delta H < \Delta V \\ \frac{\sum_{ij=24,33,35,44} e_{ij}(G_{ij} - R'_{ij})}{\sum_{ij=24,33,35,44} e_{ij}} & \text{otherwise} \end{cases} \quad (4.9)$$

where R'_{ij} is calculated by the bilinear interpolation. For example,

$$R'_{44} = \frac{R_{34} + R_{54}}{2} \quad \text{and} \quad R'_{35} = \frac{R_{34} + R_{36}}{2} \quad (4.10)$$

The definition of weighting factor e_{ij} is as same as that in Kimmel's method (3.25) [12].

The reconstruction of G channel in our method is an edge adaptive weighted interpolation scheme. Similar formula can be listed for a B centre.

4.4. R and B channel reconstruction

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

After G channel reconstruction, the missing R and B values are obtained by using weighted color difference interpolation. For instance,

$$B_{34} = G_{34} + \frac{e_{23}k_{23} + e_{25}k_{25} + e_{43}k_{43} + e_{45}k_{45}}{e_{23} + e_{25} + e_{43} + e_{45}} \quad (4.11)$$

$$B_{33} = G_{33} + \frac{e_{23}k_{23} + e_{43}k_{43} + e_{32}k_{32} + e_{34}k_{34}}{e_{23} + e_{43} + e_{32} + e_{34}} \quad (4.12)$$

where $k_{ij} = B_{ij} - G_{ij}$.

$$R_{43} = G_{43} + \frac{e_{32}k_{32} + e_{34}k_{34} + e_{52}k_{52} + e_{54}k_{54}}{e_{32} + e_{34} + e_{52} + e_{54}} \quad (4.13)$$

$$R_{33} = G_{33} + \frac{e_{23}k_{23} + e_{43}k_{43} + e_{32}k_{32} + e_{34}k_{34}}{e_{23} + e_{43} + e_{32} + e_{34}} \quad (4.14)$$

where $k_{ij} = R_{ij} - G_{ij}$.

The definition of weighting factor e_{ij} is also as same as that in Kimmel's method (3.25).

4.5 Refinement

Up to this point, the three color planes are fully populated, and the resultant values can be further updated with a refinement process. The refinement step can improve the overall accuracy of restoration because more information is available at this stage. The G channel refinement is similar to that described in Section 4.3 except for the edge detection (since all the relevant G pixels are available). For refinement of G_{34} , both the first and second directional derivatives of G color plane in a 3×3 window are used for edge detection:

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

$$\Delta H = |G_{33} - G_{35}| + |G_{33} + G_{35} - 2G_{34}| \quad (4.15)$$

$$\Delta V = |G_{24} - G_{44}| + |G_{24} + G_{44} - 2G_{34}| \quad (4.16)$$

and

$$G_{34} = R_{34} + \begin{cases} \frac{e_{24}(G_{24} - R_{24}) + e_{44}(G_{44} - R_{44})}{e_{24} + e_{44}} & \text{if } \Delta H > \Delta V \\ \frac{e_{33}(G_{33} - R_{33}) + e_{35}(G_{35} - R_{35})}{e_{33} + e_{35}} & \text{if } \Delta H < \Delta V \\ \frac{\sum_{ij=23,24,25,33,35,43,44,45} e_{ij}(G_{ij} - R_{ij})}{\sum_{ij=23,24,25,33,35,43,44,45} e_{ij}} & \text{otherwise} \end{cases} \quad (4.17)$$

The B channel is then refined using the eight neighboring pixel values in a 3×3 window, similar to (4.11):

$$B_{34} = G_{34} + \frac{\sum_{ij=23,24,25,33,35,43,44,45} e_{ij} k_{ij}}{\sum_{ij=23,24,25,33,35,43,44,45} e_{ij}} \quad (4.18)$$

where $k_{ij} = B_{ij} - G_{ij}$.

The same approach applies to R channel refinement.

4.6 Experimental results

4.6.1 Performance measure

To compare the performance of the proposed demosaicking method with the relevant existing algorithms, full color images were generated from each of Bayer pattern arrays by using different interpolation algorithms and the PSNR (Peak-Signal-to-Noise-Ratio) of these interpolated images were computed with respect to their original images.

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

Let $I_s(x, y)$, $s=1, 2, 3$ be the spectral components of an image of size $M \times N$ and $I'_s(x, y)$ be the respective reconstructed spectral components. Then PSNR is defined below:

$$PSNR = 10 \log_{10} \left(\frac{MN \cdot 255^2}{\sum_{\forall x} \sum_{\forall y} (I_s(x, y) - I'_s(x, y))^2} \right) \tag{4.19}$$

4.6.2 Experimental results

First, Figure 4.3 shows the simulated interpolation results of the test image by using different edge detectors. Figure 4.3 (a) is the original image “Ray” which contains regular and fine edges, while Figure 4.3 (b) shows a magnified version of its central portion. Figure 4.3 (c~f) illustrate the demosaicking results of the central portion, by the methods in [5], [10] and [12], and the interpolation based upon the proposed edge detector (4.3), respectively. As we can see, the three existing methods result in annoying reconstruction artifacts (false color effects) around the central area where fine edges are located, while the result of the proposed edge detector is much closer to the original image (Figure 4.3(b)). Table 4.2 echoes the results shown in Figure 4.3, by listing the associated PSNR measured with the whole images for the four methods of edge detection.

Channel	Interpolation with method [5]	Interpolation with method [10]	Interpolation with method [12]	Interpolation with (4.3)
R	30.78	31.40	29.65	40.17
G	30.83	32.18	31.36	38.08
B	30.34	31.13	29.67	40.06

Table 4.2: PSNR comparison for interpolation with different edge detectors

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

Figure 4.3: Interpolation results of *Ray* image using different edge detectors.

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

Then we compared the proposed demosaicking method comprehensively with the relevant existing algorithms for various test images. Figure 4.4 shows test images used in the experiments. These images were sampled with Bayer CFA pattern to produce the mosaic images as the input for reconstruction. In the experiments, we found Kimmel's method [12] is prone to color artifacts, in this case iterating three times made the results worse, therefore in our implementation we did color correction only once. For Gunturk et al.'s method [17], we make use of one-level decomposition with five projection iterations in our experiments.

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

Figure 4.4: Images used in experiments. (These images are referred as *Image 1* to *Image 15*, enumerated from left to right, and top to bottom, where *Image 1* is a 384×256 lighthouse image and *Image 4* is a 768×512 lighthouse image.).

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

Table 4.3 shows the PSNR comparison for the bilinear method, smooth hue transition interpolation method [4], some existing edge directed interpolation methods [5, 10, and 12], alternating projections color plane interpolation [17] and the proposed method. In most cases, our proposed method can achieve higher PSNR than other demosaicking methods. To verify it, we give the average PSNR difference between the proposed method and other relative methods in Table 4.3 as shown in Figure 4.5. The average PSNR difference in Table 4.3 is defined as

$$\text{Average PSNR difference} = \frac{\sum_{i=1}^{15} (PSNR_{i(\text{proposed method})} - PSNR_{i(\text{relative method})})}{15} \quad (4.20)$$

We also provide some examples from the images used in the experiments for the visual comparison. Figure 4.6—4.9 show magnified details of original images (*Image 1*, 4, 7 and 13 in Figure 4.4) and the corresponding reconstructed images from different demosaicking algorithms.

In Figure 4.6, it can be seen that the proposed method reduces more false color artifacts in the region around the window and the house siding than other methods. Figures 4.7—4.9 illustrate the similar visual comparisons of *Image 4*, *Image 7* and *Image 13* respectively. In the area around the fence in Figure 4.7, the region around digits in Figure 4.8, and the region around letters in Figure 4.9, color aliasing artifacts are significantly reduced by using the proposed algorithm. Further improvement is noticeable by performing just one round of refinement. It can be observed that the proposed method achieves better reconstruction in high-frequency image regions.

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

In general, the proposed method is suitable for reconstructing fine image details in the image regions that characterized by high correlation between RGB color channels. However, when encounter the image regions that the correlation between RGB components is very low, the proposed narrow edge detector (4.3) derived from different color channels will not perform very well.

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

Image No.	Channel	Bilinear method	Method in [4]	Method in [5]	Method in [10]	Method in [12] (one correction)	Method in [17] (five iterations)	Proposed Method (before and after one round of refinement)	
1	Red	24.41	27.84	31.16	32.70	32.50	35.18	33.93	35.57
	Green	29.33	29.33	32.14	34.71	36.45	39.21	35.47	38.92
	Blue	24.52	27.11	31.40	33.42	31.08	35.32	34.68	36.77
2	Red	27.99	31.17	33.62	36.08	31.56	37.76	36.96	36.85
	Green	31.85	31.85	33.75	37.69	33.51	38.71	38.32	40.38
	Blue	27.90	31.09	33.45	35.79	33.46	35.33	36.59	36.37
3	Red	21.55	24.32	26.66	30.18	27.07	31.83	31.23	32.30
	Green	25.05	25.05	26.03	30.47	26.56	32.48	31.58	33.33
	Blue	21.87	24.06	25.75	27.88	26.27	28.56	28.55	28.61
4	Red	26.75	30.57	34.40	36.71	35.94	38.71	37.39	39.07
	Green	31.71	31.71	34.95	38.39	39.44	42.55	38.60	42.09
	Blue	26.92	29.97	34.53	37.18	34.37	38.98	37.98	39.77
5	Red	25.63	27.94	30.18	31.96	28.83	32.46	32.40	31.52
	Green	29.33	29.33	31.06	35.82	32.10	36.18	36.26	37.37
	Blue	25.69	29.18	31.49	35.10	32.13	33.19	35.45	34.49
6	Red	26.53	28.95	30.52	32.11	31.66	33.58	32.54	33.46
	Green	31.71	31.71	32.93	35.94	29.69	38.51	36.16	38.93
	Blue	28.11	31.26	32.70	34.77	28.15	36.03	35.57	36.80
7	Red	28.00	30.97	34.36	36.52	35.55	37.29	37.59	38.42
	Green	32.42	32.42	34.93	38.63	39.84	40.54	39.22	41.99
	Blue	28.01	31.01	34.80	37.73	35.77	37.32	38.40	38.66
8	Red	26.41	28.78	30.76	32.49	31.56	34.16	32.97	33.58
	Green	30.38	30.38	31.74	34.85	36.11	37.22	35.25	38.38
	Blue	26.70	29.64	31.42	33.44	33.04	34.03	34.33	35.48
9	Red	31.68	34.87	37.04	39.87	38.53	41.61	40.53	42.12
	Green	35.21	35.21	36.92	40.63	41.18	44.47	40.98	44.34
	Blue	31.58	34.24	36.02	38.14	38.77	40.61	38.93	40.13
10	Red	24.03	28.32	30.47	31.10	33.83	33.92	34.97	40.27
	Green	28.78	28.78	30.53	31.88	35.27	37.77	34.20	40.96
	Blue	23.68	27.49	30.04	30.82	30.89	33.78	35.17	39.78
11	Red	26.57	23.06	31.46	33.48	33.65	35.22	34.28	35.04
	Green	31.08	31.08	32.68	36.07	36.35	38.38	36.65	39.58
	Blue	27.37	28.71	32.48	34.54	33.58	35.12	35.50	36.52
12	Red	23.58	26.29	28.74	30.42	30.14	31.36	31.06	30.98
	Green	27.95	27.95	30.09	33.25	32.13	34.07	33.99	36.47
	Blue	22.88	25.64	28.36	30.19	28.27	29.57	31.04	31.45
13	Red	32.70	35.87	38.15	40.44	38.30	41.17	40.64	41.10
	Green	37.14	37.14	39.00	42.41	41.75	43.59	42.33	44.23
	Blue	32.95	34.50	37.62	39.72	34.57	40.61	39.79	39.09
14	Red	19.14	22.19	25.18	26.91	27.28	28.79	29.15	31.87
	Green	24.39	24.39	26.40	28.60	31.19	33.20	30.29	34.14
	Blue	19.28	21.94	25.18	26.85	26.47	28.67	29.06	31.67
15	Red	24.03	28.32	30.47	31.10	33.83	33.92	34.97	40.27
	Green	28.78	28.78	30.53	31.88	35.27	37.77	34.20	40.96
	Blue	23.68	27.49	30.04	30.82	30.89	33.78	35.17	39.78

Table 4.3: PSNR comparison of different demosaicking methods

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

Figure 4.5: Average PSNR difference between the proposed method and other relative methods in Table 4.3.

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

(a) Magnified details of the original image

(b) Bilinear interpolation

(c) Laroche and Prescott's method

(d) Hamilton and Adams' method

Figure 4.6: Visual comparison of *Image 1* (384×256 lighthouse image).

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

(e) Kimmel's method

(f) Gunturk et al.'s method

(g) Proposed method (before refinement)

(h) Proposed method (after one round of refinement)

Figure 4.6: (Continued.) Visual comparison of Image 1 (384×256 lighthouse image).

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

Figure 4.7: Visual comparison of *Image 4* (768×512 lighthouse image).

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

(e) Kimmel's method

(f) Gunturk et al.'s method

(g) Proposed method (before refinement)

(h) Proposed method (after one round of refinement)

Figure 4.7: (Continued.) Visual comparison of Image 4 (768 × 512 lighthouse image).

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

(a) Magnified details of the original image

(b) Bilinear interpolation

(c) Laroche and Prescott's method

(d) Hamilton and Adams' method

Figure 4.8: Visual comparison of *Image7*.

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

(e) Kimmel's method

(f) Gunturk et al.'s method

(g) Proposed method (before refinement)

(h) Proposed method (after one round of refinement)

Figure 4.8: (Continued.) Visual comparison of Image 7.

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

(a) Magnified details of the original image

(b) Bilinear interpolation

(c) Laroche and Prescott's method

(d) Hamilton and Adams' method

Figure 4.9: Visual comparison of *Image 13*.

Chapter 4 Color Demosaicking with Improved Edge-Direction Detection

(e) Kimmel's method

(f) Gunturk et al.'s method

(g) Proposed method (before refinement)

(h) Proposed method (after one round of refinement)

Figure 4.9: (Continued.) Visual comparison of Image 13.

Chapter 5

Ringling Artifacts Reduction in JPEG Compressed Digital Images

In a digital still camera, the image is captured, processed, and stored. After demosaicking, the resulting image will be compressed to store in the camera in order to maximize the efficiency of storage. Since the compression process produces artifacts that degrade the image visual quality, to improve the visual quality of the final image taken by the digital still camera, the needs for post-processing schemes for compressed images are essential.

5.1 Image compression in digital still cameras

Color digital images acquired by a digital camera are typically mosaiced. To store a large number of images in its finite memory space, one can either store the raw data values, compress the data to store using some lossless compression technique, or to demosaic the data on the camera and then compress the full color images using a lossy scheme [52]. Although lossless compression schemes can achieve some saving of the storage required, for larger compression ratios it is usually necessary to use a lossy compression method, each camera is therefore equipped with a compression engine, and JPEG (Joint Photographic Experts Group) [29] is typical of such compression schemes. While the decoding is done after the images are downloaded to a computer host.

Chapter 5 Ringing Artifacts Reduction in JPEG Compressed Digital Images

JPEG compression standards are based on block discrete cosine transform (BDCT), and BDCT image coding techniques are widely used in image compression application due to its near-optimum energy compaction property and availability of fast algorithms and hardware. At a moderate bit rate, BDCT is usually a quiet satisfactory solution to most of practical coding applications. However, when the bit rate becomes low, visual annoying artifacts appear in decoded images. The two most noticeable JPEG compression artifacts are blocking in flat regions and ringing along object edges.

5.1.1 JPEG still picture compression standard

This section describes the basic factors that lead to distortions in the JPEG compression process. A joint ISO/CCITT committee known as JPEG (Joint Photographic Experts Group) has been working to establish the first international compression standard for continuous-tone still images, both grayscale and color. There are a lot of books and papers written on JPEG [29], [30], here we will only describe the principal ideas.

For color image compression, the input RGB image is processed in the following steps:

1). Color transformation: Each pixel RGB-triplet is transformed into YCbCr color space, where Y represents luminance, Cb and Cr represent chrominance. YCbCr (256 levels) can be computed directly from 8-bit RGB as follows:

$$\begin{aligned} Y &= 0.299R + 0.587G + 0.114B \\ C_b &= -0.1687R - 0.3313G + 0.5B + 128 \\ C_r &= 0.5R - 0.4187G - 0.0813B + 128 \end{aligned} \tag{5.1}$$

Chapter 5 Ringing Artifacts Reduction in JPEG Compressed Digital Images

2). Sub-sampling the chrominance spaces: Since the human visual system has less acuity in chrominance than in luminance, the Cb, Cr spaces are sampled by a factor of 2 in each axis.

3). Partitioning: Each of the YCbCr color planes is partitioned into disjoint 8×8 blocks.

Each of these blocks is then compressed as shown in Figure 5.1.

Figure 5.1: DCT-Based encoder processing steps

To reverse the compression and display a close approximation to the original image, the compressed data is fed into the reverse process as shown in Figure 5.2.

Figure 5.2: DCT-Based decoder processing steps

Chapter 5 Ringing Artifacts Reduction in JPEG Compressed Digital Images

- FDCT and IDCT

At the input to the encoder in Figure 5.1, source image samples are grouped into 8×8 blocks, shifted from an unsigned integer with range $[0, 2^{p-1} - 1]$ to a signed integer with range $[-(2^{p-1}), 2^{p-1} - 1]$ by subtracting 2^{p-1} from the value where p is the number of bits per channel (or bits per value). In the case of the standard 8-bit channel the numbers are shifted from $[0, 255]$ to $[-128, 127]$ by subtracting 128, and input to the Forward DCT (5.2). At the output from the decoder, the Inverse DCT (5.3) outputs 8×8 sample blocks to form the reconstructed image.

$$FDCT(u, v) = \frac{1}{4} C(u) C(v) \left[\sum_{x=0}^7 \sum_{y=0}^7 p(x, y) * \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16} \right] \quad (5.2)$$

$$IDCT(x, y) = \frac{1}{4} \left[\sum_{u=0}^7 \sum_{v=0}^7 C(u) C(v) FDCT(u, v) * \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16} \right] \quad (5.3)$$

where

$$C(u), C(v) = \begin{cases} \frac{1}{\sqrt{2}} & \text{if } u, v = 0 \\ 1 & \text{otherwise} \end{cases} \quad (5.4)$$

and $p(x, y)$ is the matrix of values from the original image.

The output of the FDCT is the set of 64 DCT coefficients, the DCT coefficient values can be regarded as the relative amount of the 2D spatial frequencies contained in the 64-point input signal. The coefficient with zero frequency in both dimensions is called the “DC coefficient” and the rest are called “AC coefficients”.

Chapter 5 Ringing Artifacts Reduction in JPEG Compressed Digital Images

- Quantization

Quantization is the step where we actually throw away data. The DCT is a lossless procedure. The data can be precisely recovered through the IDCT. During Quantization every element in the 8x8 FDCT matrix is divided by a corresponding element in a quantization matrix Q to yield a matrix Q_{FDCT} according to the formula:

$$Q_{FDCT} = \text{round}\left(\frac{FDCT(u,v)}{Q(u,v)}\right) \quad (5.5)$$

The goal of quantization is to reduce most of the less important high frequency coefficients to zero, the more zeros we can generate the better the image will compress. The matrix Q generally has lower numbers in the upper left and increase in magnitude as they get closer to the lower right. It means higher frequency coefficients are quantized using larger quantization step.

Dequantization is the inverse function, which returns the result to a representation appropriate for input to the IDCT:

$$FDCT(u,v) = Q_{FDCT} * Q(u,v) \quad (5.6)$$

Those values that were reduced to zero are gone forever and cannot be reconstructed.

- Entropy Encoding

After quantization, all of the quantized coefficients are ordered into the “zig-zag” sequence, shown in Figure 5.3. This ordering tends to group the lower coefficients in the

beginning of the string and distribute the zeros in longer continuous strings. The strings are then passed on to a Huffman or arithmetic encoder. These further condense the string by looking for recurring sequences of numbers that are placed into a dictionary and referenced. The data is then sent to a receiving computer that performs the reverse operation on the data and reconstructs the image.

Figure 5.3: The zig-zag pattern of Entropy Encoding

5.1.2 Compression artifacts

JPEG compression standard is based on block discrete cosine transform (BDCT), and when the bit rate becomes low, visually annoying *blocking* and *ringing* artifacts appear in the decompressed images.

Blocking artifacts appear as intensity discontinuity between two adjacent blocks, due to quantizing low-frequency coefficients in each block without considering the correlation between adjacent blocks.

Chapter 5 Ringing Artifacts Reduction in JPEG Compressed Digital Images

The ringing artifacts are caused by a coarse quantization of high frequency components. When severely quantized, high frequency components are filtered out. Through the inverse transform, the quantization error appears as ringing noise that observable as periodic pseudo-edges around original sharp edges and cause severe visual degradation. Depending on underlying compression scheme's properties, the ringing artifacts will spread out to a well defined finite extent region surrounding the edges. The ripples due to ringing artifact are strongest in regions very close to edges and they become weaker when their position gets farther from the edge.

5.2 Proposed spatial adaptive de-ringing algorithm

Many post-processing methods [31-57] have been developed for reducing the visual artifacts in compressed images. Among these methods are adaptive spatial filtering methods [31-36], wavelet-based filtering methods [37], [38], DCT-domain methods [39], [40], and projection onto convex sets methods [41], [42] and [43].

Since the compression artifacts resulting from block DCT coding are high frequency artifacts, a straightforward solution is to apply low-pass filtering to the region where artifacts occur. In this work, we propose a spatial adaptive post-processing algorithm focusing on ringing artifacts reduction by exploiting the local statistics of the JPEG compressed image.

The human visual system is very sensitive to high frequency changes, especially to the

Chapter 5 Ringing Artifacts Reduction in JPEG Compressed Digital Images

edges in the image. In order to preserve the edges and other strong fine details (e.g. texture) in the JPEG compressed image, the basic idea of our de-ringing algorithm is to classify the image pixels before performing the filtering. In our method, we only process the Y (luminance) band in the YCbCr format in a JPEG compressed image.

5.2.1 Pixel classification

Since the local variance (5.7) reflects the gradient of the image at each pixel location, we can classify each pixel as edge, texture, uniform or ringing according to the local variance value of each pixel.

$$\sigma_1^2 = \sum_{i=1}^M \sum_{j=1}^N \frac{(X_{i,j} - \bar{X})^2}{M * N} \quad (5.7)$$

where: i and j : row and column pixel index,

M and N : row and column size of the window,

$X_{i,j}$: value of the pixel at position (i, j) ,

$\bar{X}$: mean value of all the pixels in the window.

In our algorithm, first we compute the local (3×3) variance $\sigma_1^2(x, y)$ at each pixel and classify the pixel type via

$$pixel(x, y) = \begin{cases} \text{uniform}, & \sigma_1^2(x, y) \leq T_u; \\ \text{texture or ringing}, & T_u < \sigma_1^2(x, y) < T_e; \\ \text{edge}, & T_e < \sigma_1^2(x, y). \end{cases} \quad (5.8)$$

where T_u and T_e are thresholds set to 20 and 400 experimentally, respectively.

Chapter 5 Ringing Artifacts Reduction in JPEG Compressed Digital Images

The low variance ($\sigma_1^2(x, y) \leq T_u$) corresponds to the smooth region, the high variance ($\sigma_1^2(x, y) > T_e$) corresponds to the edges in the image, and the pixels with the variance between these two thresholds can be classified as either texture or ringing artifacts depending on the neighboring pixels.

Since the ringing artifacts and the texture pixels have similar local variance $\sigma_1^2(x, y)$ value in a 3×3 window, we use the variance $\sigma_2^2(x, y)$ in a larger 15×15 window to differentiate them. The reason we choose a 15×15 window is that ringing occurs near edges and independently between 8×8 blocks in the JPEG compressed image, which means when a strong or middle ringing noise is at the center of a 15×15 window, it must have some neighboring edge pixels within this window, in this case there will be a relatively high variance value $\sigma_2^2(x, y)$. Otherwise the relatively low variance $\sigma_2^2(x, y)$ will correspond to the texture pixel or ringing noise around the texture region since there are no strong edge pixels around it.

Additionally, according to the value of local variance $\sigma_1^2(x, y)$, the ringing or texture region will be further classified as two groups, group one is strong ringing or texture region, group two is middle ringing or texture region, and the ringing detection thresholds T_{r1} and T_{r2} for these two groups will be selected separately for differentiating the ringing noises surrounding the edges from the texture region more accurately. Taking these into account, we detect ringing artifacts as follows:

Chapter 5 Ringing Artifacts Reduction in JPEG Compressed Digital Images

- Group 1:

For $t < \sigma_1^2(x, y) < T_e$ (strong ringing artifacts or texture region),

$$pixel(x, y) = \begin{cases} \text{strong ringing surrounding the edges,} & T_{r1} < \sigma_2^2(x, y); \\ \text{strong texture / ringing in the texture region,} & \sigma_2^2(x, y) < T_{r1}. \end{cases} \quad (5.9)$$

- Group 2:

For $T_u < \sigma_1^2(x, y) < t$ (middle ringing artifacts or texture region),

$$pixel(x, y) = \begin{cases} \text{middle ringing surrounding the edges,} & T_{r2} < \sigma_2^2(x, y); \\ \text{middle texture / ringing in the texture region,} & \sigma_2^2(x, y) < T_{r2}. \end{cases} \quad (5.10)$$

where t , T_{r1} and T_{r2} are determined experimentally and set to 100, 800 and 500 respectively.

5.2.2 De-ringing

After the pixel classification, pixels can be labeled as uniform, edge, texture, ringing in the texture region or ringing surrounding the edges, the de-ringing operation left the uniform and edge pixels unfiltered and goes to the ringing and texture pixels directly. For strong or middle ringing surrounding the edges, we use a 3×3 median filter to remove these ringing artifacts. For texture or ringing artifacts in the texture region, we use an edge adaptive low-pass filter to perform the post-filtering, since it can both keep the texture information because of its edge-adaptive characteristic and reduce the ringing artifacts as it is a low-pass filter.

Y1	Y2	Y3
Y4	Y5	Y6
Y7	Y8	Y9

Figure 5.4: Luminance (Y) values in a 3×3 window.

According to Figure 5.4, if Y5 is a non-edge, non-uniform pixel, the following filter will be applied.

$$Y5 = \begin{cases} \frac{Y2 + Y5 + Y8}{3} & \text{if } \Delta H > \Delta V \\ \frac{Y4 + Y5 + Y6}{3} & \text{if } \Delta H < \Delta V \\ 3 \times 3 \text{ median filter} & \text{otherwise} \end{cases} \quad (5.11)$$

where ΔH and ΔV are defined as follows:

$$\Delta H = |Y4 - Y6| + |Y4 + Y6 - 2Y5| \quad (5.12)$$

$$\Delta V = |Y2 - Y8| + |Y2 + Y8 - 2Y5| \quad (5.13)$$

Since it is a directional low-pass filter, it can reduce the ringing artifacts by low-pass filtering and meanwhile help to preserve the texture information by direction estimation that prevent over smoothing the details.

5.3 Experimental results

The proposed algorithm is applied to JPEG compressed images with ringing artifacts. The test images are divided into blocks of size 8×8 for JPEG compression and the quantization table used in our experiment is shown as Table 5.1.

Chapter 5 Ringing Artifacts Reduction in JPEG Compressed Digital Images

20	24	28	32	36	80	98	144
24	24	28	34	52	70	128	184
28	28	32	48	74	114	156	190
32	34	48	58	112	128	174	196
36	52	74	112	136	162	206	224
80	70	114	128	162	208	242	200
98	128	156	174	206	242	240	206
144	184	190	196	224	200	206	208

Table 5.1: Quantization table used in the experiment.

Figure 5.5 shows detected ringing artifacts around edges in test images. As we can see most ringing artifacts around edges (black pixels for emphasizing) can be differentiated from texture pixels (lawn in *cameraman* image and water in *sailboat* image) by using our pixel classification scheme (5.9 and 5.10), thus prevent over smoothing the texture region by post-filtering the ringing around edges and texture region separately.

Figure 5.5: Detected ringing artifacts around edges

Chapter 5 Ringing Artifacts Reduction in JPEG Compressed Digital Images

Table 5.2 shows the PSNR comparison of luminance (Y) between JPEG decoded and post processed images.

Image	JPEG	After de-ringing
Cameraman	28.51	28.84
Sailboat	31.69	31.95
Plane	34.38	34.55
Lighthouse	31.59	31.67
Bike	28.90	29.18
Actor	29.25	29.42

Table 5.2: PSNR comparison of luminance (Y) between the JPEG decoded and post processed images.

A portion of the luminance (Y) of decode *cameraman*, *plane* and *sailboat* images are shown in Figure 5.6(a), Figure 5.7(a) and Figure 5.8(a), the results by using the proposed de-ringing algorithm are shown in Figure 5.6(b), Figure 5.7(b) and Figure 5.8(b).

Chapter 5 Ringing Artifacts Reduction in JPEG Compressed Digital Images

(a) JPEG compressed

(b) After de-ringing

Figure 5.6: The luminance of JPEG compressed *cameraman* image: before and after de-ringing post processing.

Chapter 5 Ringing Artifacts Reduction in JPEG Compressed Digital Images

(a) JPEG compressed

(b) After de-ringing

Figure 5.7: The luminance of JPEG compressed *plane* image: before and after de-ringing post processing.

Chapter 5 Ringing Artifacts Reduction in JPEG Compressed Digital Images

Figure 5.8: The luminance of JPEG compressed *sailboat* image: before and after de-ringing post processing.

Chapter 6

Conclusion and Future Recommendation

In this thesis, we investigate and develop the image processing techniques, demosaicking and JPEG compressed image post-processing, in a digital photography system. Some of the well known demosaicking methods are discussed. An improved edge-adaptive color demosaicking method for Bayer pattern in single-sensor digital still cameras is proposed. The effort has been directed in more accurate edge direction detection in fine image details, emphasis on G channel restoration and overall refinement step by making full use of the intra- and inter-channel correlations. Experimental results show that the proposed demosaicking algorithm can reconstruct edges and fine image details better, therefore reduce color aliasing artifacts in these regions. A new spatial adaptive post-processing algorithm focusing on ringing artifacts reduction by exploiting the local statistics of the JPEG compressed image has also been proposed. The proposed de-ringing scheme can reduce ringing artifacts around edges and prevent over smoothing texture regions in JPEG compressed images.

It is concluded that the visual quality of the images taken by the digital still camera can be improved by proposed demosaicking method for more accurate image reconstruction

Chapter 6 Conclusions

and de-ringing post processing scheme to reduce JPEG compression artifacts.

For further exploration of demosaicking, by making use of the assumption that the color in the picture makes relatively smooth transitions, we may use a larger area of the CCD to estimate the missing color.

Bibliography

- [1] R. C. Gonzalez, R. E. Woods, *Digital Image Processing*, Second Edition, Prentice-Hall, 2002.
- [2] Y. Okano, et al. "Electronic digital still camera using 3-CCD image sensors", *Proceedings of the IS&T's 48th Annual Conference*, pp.428-432, 1995.
- [3] B. E. Bayer, "Color imaging array," *U.S. Patent 3,971,065*, Jul. 1976.
- [4] 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*, 1987.
- [5] 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*, 1994.
- [6] D. H. Brainard, "Bayesian method for reconstructing color images from trichromatic samples," *Proceedings of the IS&T's 47th Annual Conference*, pp. 375-380, 1994.
- [7] R. H. Hibbard, "Apparatus and method for adaptively interpolating a full color image utilizing luminance gradients," *U.S. Patent 5,382,976*, 1995.
- [8] J. E. Adams Jr., "Interactions between color plane interpolation and other image processing functions in electronic photography," *Proceeding of SPIE, Cameras and Systems for Electronic Photography and Scientific Imaging*, vol. 2416, pp. 144-151, Feb. 1995.

Bibliography

- [9] D. H. Brainard, and D. Sherman, "Reconstructing images from trichromatic samples: from basic research to practical applications," *Proceedings of the 3rd IS&T/SID Color Imaging Conference*, pp. 4-10, 1995.
- [10] J. F. Hamilton Jr. and J. E. Adams, "Adaptive color plane interpolation in single sensor color electronic camera," *U.S. Patent 5,629,734*, 1997.
- [11] J. E. Adams, Jr. "Design of practical color filter array interpolation algorithms for digital cameras," *Proceeding of SPIE*, vol. 3028, pp. 117-125, 1997.
- [12] R. Kimmel, "Demosaiicing: image reconstruction from CCD samples," *IEEE Trans. Image Processing*, vol. 8, pp. 1221-1228, 1999.
- [13] B. Lee, J. Kim and C. Lee, "High quality image interpolation for color filter arrays," *Systems Man, and Cybernetics, 2000 IEEE International Conference*, vol.2, pp.1547 –1550, 2000.
- [14] S. C. Pei and I. K. Tam; "Effective color interpolation in CCD color filter array using signal correlation," *Image Processing, 2000. Proceedings, 2000 International Conference*, vol.3, pp. 488–491, 2000.
- [15] W. Lu and Y. P. Tan, "Layering-based color filter array interpolation," *Image Processing, 2001. Proceedings. 2001 International Conference*, vol. 3, pp. 860-863, 2001.
- [16] J. W. Glotzbach, R. W. Schafer and K. Illgner, "A method of color filter array interpolation with alias cancellation properties," *Image Processing, 2001.Proceedings. 2001 International Conference*, vol.1, pp. 141 -144, 2001.
- [17] B. K. Gunturk, Y. Altunbasak and R. M. Mersereau, " Color plane interpolation

Bibliography

- using alternating projections,” *IEEE Trans. Image Processing*, vol. 11, pp. 997-1013, Sep. 2002.
- [18] Freeman, T. William, “Median filter for reconstructing missing color samples”, *United States Patent* 4,724,395, 1988
- [19] D. D. Muresan and T. W. Parks “*Optimal Recovery Demosaicing*,” *IASTED Signal and Image Processing*, 2002.
- [20] B. S. Hur and M. G. Kang; “High definition color interpolation scheme for progressive scan CCD image sensor,” *IEEE Trans. Consumer Electronics*, vol. 47, pp.179 –186, Feb. 2001.
- [21] J. Mukherjee, M. S. Moore and S. K. Mitra, “Color demosaicing with constrained buffering,” *Signal Processing and its Applications, Sixth International Symposium*, vol.1, pp. 52 -55, Aug. 2001.
- [22] C. Weerasinghe, I. Kharitonenko and P. Ogunbona, “Method of color interpolation in a single sensor color camera using green channel separation,” *Acoustics, Speech, and Signal Processing, 2002 IEEE International Conference*, vol.4, pp. IV-3233 -IV-3236, 2002.
- [23] T. Kuno, H. Sugiura, and N. Matoba, “New interpolation method using discriminated color correlation for digital still camera,” *IEEE Trans. Consumer Electronics*, vol. 45, no. 1, pp. 259-267, Feb, 1999.
- [24] R. Kakarala and Z. Baharav, “Adaptive demosaicing with the principal vector method,” *IEEE Trans. Consumer Electronics*, vol. 48, no. 4, pp. 932-937, Nov, 2002.

Bibliography

- [25] W. Lu and Y.-P. Tan, "Color filter array demosaicking: new method and performance measures," *IEEE Trans. Image Processing*, vol. 12, no. 10, pp. 1194-1210, Oct, 2003.
- [26] R. Lukac, K. N. Plataniotis, D. Hatzinakos and M. Aleksic, "A novel cost effective demosaicing approach," *IEEE Trans. Consumer Electronics*, vol. 50, no. 1, pp. 256-261, Feb, 2004.
- [27] C. Cai, T.-H. Yu and S. K. Mitra, "Saturation-based adaptive inverse gradient interpolation for Bayer pattern images," *Vision, Image and Signal Processing, IEE Proceeding-*, vol.148, issue. 3, pp. 202 – 208, Jun. 2001
- [28] Longere, P.; Xuemei Zhang; Delahunt, P.B.; Brainard, D.H., "Perceptual assessment of demosaicing algorithm performance" *Proceedings of the IEEE*, vol. 90, issue. 1, pp. 123 –132, Jan. 2002.
- [29] G. Wallace, "The JPEG still-picture compression standard," *Communications of the ACM*, vol.34, pp. 31-46, Apr. 1991 .
- [30] W. B. Pennebaker and J. L. Mitchell, *JPEG still image data compression standard*, Van Nostrand Reinhold, New York, 1993.
- [31] W. E. Lynch, A. R. Reibman, and B. Liu, "Postprocessing transform coded images using edges," *Proc. IEEE ICASSP*, pp. 2323-2326, May 1995.
- [32] C. J. Kuo and R. J. Hsieh, "Adaptive postprocessing for block encoded images," *IEEE Trans. Circuits Syst. Video Technol.*, vol.5, no.4, pp. 298 – 304, Aug. 1995.
- [33] A. Kundu, "Enhancement of JPEG coded images by adaptive spatial filtering," *Proc, IEEE International Conference on Image Processing*, pp.187-190,

Bibliography

Oct.1995.

- [34] J. Hu, N. Sinaceur, F. Li, K. W. Tam and Z. Fan, "Removal of blocking and ringing artifacts in transform coded images," *Proc. IEEE ICASSP*, vol. 4, pp. 2526-2568, Apr.1997.
- [35] M.-Y. Shen and C.-C. J. Kuo, "Review of postprocessing techniques for compression artifacts removal," *Journal of Visual Communication and Image Representation*, vol. 9, no. 1, pp. 2-14, Mar.1998.
- [36] H. S. Kong, A. Vetro and H. Sun, "Edge map guided adaptive post-filter for blocking and ringing artifacts removal," *Circuits and Systems, Proceedings of the International Symposium on*, vol.3, pp. 929-32. May 2004.
- [37] Z. Xiong, M. T. Orchard, and Y. Q. Zhang, "A deblocking algorithm for JPEG compressed images using overcomplete wavelet representations," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 2, pp. 433-437, Aug. 1997.
- [38] A. W.-C. Liew and H. Yan, "Blocking artifacts suppression in block-coded images using overcomplete wavelet representation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no.4, pp. 450 – 461, Apr. 2004.
- [39] T. Chen, H. R. Wu and B. Qiu, "Adaptive post-filtering of transform coefficients for the reduction of blocking artifacts," *IEEE Trans. Circuits Syst. Video Technol.*, vol.11, no. 5, pp. 594-602, May 2001.
- [40] S. Liu and A. C. Bovik, "Efficient DCT-domain blind measurement and reduction of blocking artifacts," *IEEE Trans. Circuits Syst. Video Technol.*, vol.12, no. 12, pp. 1139-1149, Dec. 2002.

Bibliography

- [41] A. Zakhor, "Iterative procedures for reduction of blocking effects in transform image coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 2, no. 1, pp. 91-95, Mar. 1992.
- [42] J. K. Su and R. M. Mersereau, "Post-processing for artifact reduction in JPEG-compressed images," *Proc. IEEE ICASSP*, vol. 4, pp. 2363-2366, May 1995.
- [43] C. Weerasinghe, A. W.-C. Liew and H. Yan, "Artifact reduction in compressed images based on region homogeneity constraints using the projection onto convex sets algorithm," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 10, pp. 891-897, Oct. 2002.
- [44] R. Eschbach, "Decompression of standard ADCT-compressed images," *U.S. Patent 5,379,122*, 1995.
- [45] Y. K. Lai, J. Li and C.C.J. Kuo, "Image enhancement for low bit-rate JPEG and MPEG coding via postprocessing," *Visual Communications and Image Processing '96. Proc. SPIE*, vol. 2727, pp.1484-1494, Mar. 1996.
- [46] G. Lakhani, "Improved equations for JPEG's blocking artifacts reduction approach," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 6, pp. 930-934, Dec. 1997.
- [47] Y. L. Lee, H. C. Kim and H.W. Park, "Blocking effect reduction of JPEG images by signal adaptive filtering," *IEEE Trans. Image Processing.*, vol. 7, no. 2, pp. 229-234, Feb. 1998
- [48] J. Mateos, C. Ilia, B. Jimenez, R. Molina and A. K. Katsaggelos, "Reduction of

Bibliography

- blocking artifacts in block transformed compressed color images,” *International Conference on Image Processing, Proceedings*. vol. 1, pp. 401 – 405, Oct. 1998.
- [49] J. Chou, M. Crouse and K. Ramchandran, “A simple algorithm for removing blocking artifacts in block-transform coded images,” *Signal Processing Letters, IEEE*, vol. 5, issue. 2, pp.33 – 35, Feb. 1998.
- [50] C. Herley, “A post-processing algorithm for compressed digital camera images,” *International Conference on Image Processing, Proceedings*. vol. 1, pp. 396 – 400, Oct. 1998.
- [51] S. H. Oguz, Y. H. Hu and T. Q. Nguyen, “Image coding ringing artifact reduction using morphological post-filtering,” *IEEE Second Workshop on Multimedia Signal Processing*, pp. 628 – 633, Dec. 1998.
- [52] C. Herley, “Artifact reduction compression method and apparatus for mosaiced images,” *U.S. Patent 5,838,818*, 1998.
- [53] C. Herley, “Artifact reduction decompression method and apparatus for interpolated images,” *U.S. Patent 6,044,177*, 2000.
- [54] A. Nosratinia, “Enhancement of JPEG-compressed images by re-application of JPEG,” *Special Issue on Multimedia Signal Processing, Journal of VLSI Signal Processing Systems*. vol. 27, issue 1-2, pp.69 - 79, Feb. 2001.
- [55] Y.-C. Liaw, W. Lo and Z. C. Lai, “Image restoration of JPEG coded images using mean removed classified vector quantization,” *IEEE International Symposium on Circuits and Systems. ISCAS 2001*. vol. 2, pp. 509 – 512, May 2001.
- [56] S.-C. Tai, C.-C. Wang, L.-S. Huang and Y.-R. Chen, “Morphological de-ringing

Bibliography

- filter design for JPEG-2000,” *Journal of Information Science and Engineering* 19, pp. 667-680, 2003.
- [57] E. Y. Lam, “Image restoration in digital photography,” *IEEE Trans. Consumer Electronics*. vol. 49, issue. 2, pp. 269 – 274, May 2003.
- [58] P. W. Wong, D. Tretter, C. Herley, N. Moayeri and R. Lee, “Image processing considerations for digital photography,” *Compcon '97. Proceedings, IEEE*, pp. 280 – 285, Feb. 1997.
- [59] K. Parulski and M. Rabbani, “The continuing evolution of digital cameras and digital photography systems,” *IEEE International Symposium on Circuits and Systems. ISCAS 2000*. vol. 5, pp. 101 – 104, May 2000.