Image stitching and object insertion in the gradient domain

by

Ioana Speranţa Sevcenco
B.Sc., University of Bucharest, 2002
M.Sc., University of Bucharest, 2004

A Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of

MASTER OF APPLIED SCIENCE

in the Department of Electrical and Computer Engineering

© Ioana Speranţa Sevcenco, 2011
University of Victoria

All rights reserved. This thesis may not be reproduced in whole or in part, by photocopying or other means, without the permission of the author.

Image stitching and object insertion in the gradient domain

by

Ioana Speranţa Sevcenco
B.Sc., University of Bucharest, 2002
M.Sc., University of Bucharest, 2004

Supervisory Committee

Dr. Pan Agathoklis, Supervisor
(Department of Electrical and Computer Engineering)

Dr. Wu-Sheng Lu, Departmental Member
(Department of Electrical and Computer Engineering)

Supervisory Committee

Dr. Pan Agathoklis, Supervisor
(Department of Electrical and Computer Engineering)

Dr. Wu-Sheng Lu, Departmental Member
(Department of Electrical and Computer Engineering)

ABSTRACT

In this thesis, the applications of image stitching and object insertion are considered and two gradient based approaches offering solutions are proposed. An essential part of the proposed methods is obtaining an image from a given gradient data set. This is done using an existing Haar wavelet based reconstruction technique, which consists of two main steps. First, the Haar wavelet decomposition of the image to be reconstructed is obtained directly from a given gradient. Second, the image is obtained using Haar wavelet synthesis. In both stitching and object insertion applications considered, the gradient from which the image must be reconstructed is a non-conservative vector field and this requires adding an iterative Poisson solver at each resolution level, during the synthesis step of the reconstruction technique. The performance of the reconstruction algorithm is evaluated by comparing it with other existing techniques, in terms of solution accuracy and computation speed.

The proposed image stitching technique consists of three main parts: registering the images to be combined, blending their gradients over a region of interest and obtaining a composite image from a gradient. The object insertion technique considers the images registered and has two main stages: gradient blending of images in a region of interest and recovering an image from the gradient.

The performance of the stitching algorithm is evaluated visually, by presenting the results produced to combine images with varying orientation, scales, illumination, and color conditions. Experimental results illustrate both the stitching and the insertion techniques proposed, and indicate that they yield seamless composite images.

Table of Contents

List of Tables

List of Figures

List of Acronyms

1D one-dimensional

2D two-dimensional

3D three-dimensional

DCT discrete cosine transform

DFT discrete Fourier transform

MP megapixel

RGB red green blue

SNR signal to noise ratio

YC_bC_r luminance, blue and red chrominance

ACKNOWLEDGEMENTS

I often tell my colleagues how fortunate I am to be working under the supervision of Dr. Pan Agathoklis and now is the time to directly thank my supervisor for his ongoing guidance, patience and support. Besides sharing his expertise, Dr. Agathoklis has constantly strived to make me look beyond the equations into the true meaning of engineering concepts. The ease with which Dr. Agathoklis always points precisely towards the essential never ceases to amaze me and his question "What is the most important thing?" has guided me through the chapters of this thesis and beyond.

Next, I would like to express heartfelt gratitude towards another extraordinary professor: Dr. Wu-Sheng Lu. I have sat in four courses taught by Dr. Lu since I began my program at the University of Victoria and his knowledge and endless enthusiasm have impressed me each and every lecture. I will always have Dr. Lu's enthusiasm and passion for teaching in the back of my mind in my future work with my students.

Along the way, at the University of Victoria I met a number of wonderful people who to some extent changed me into what I am today: all my students, Dr. Mihai Sima, Ms. Vicky Smith, Ms. Lynne Barrett, Ms. Moneca Bracken and Ms. Janice Closson. I would also like to thank the IT department for their assistance and instructive conversations which helped me gain an insight into the technical side of my teaching or research activity. I should also thank my colleague Chamira Edussooriya, for helping me overcome some LaTeXchallenges and for always greeting me with a smile. Sincere appreciation also goes to Dr. Peter Hampton, for his support shown during many enlightening talks concerning an important part of this thesis.

A significant role in the existence of this thesis pertains to my family. It is rightful to thank my beloved mother, my brother, and my sister-in-law, for convincing me to come to Canada and continue my education, my step-father, for believing in me and showing permanent support for all my decisions, and my father, for financial support and guidance. I would also be remiss if I did not thank the ones I often call my Canadian parents, Barbara and Monte, for being by my side along this journey. I owe most of this to all of you.

Lastly, I would like to thank the Natural Sciences and Engineering Research Council of Canada and the University of Victoria for the financial support that helped me throughout my studies, and my M. Sc. advisor, Dr. Cristian Voica from the University of Bucharest, for encouraging me to pursue this endeavour.

Chapter 1

Introduction

1.1 Image processing in the gradient domain

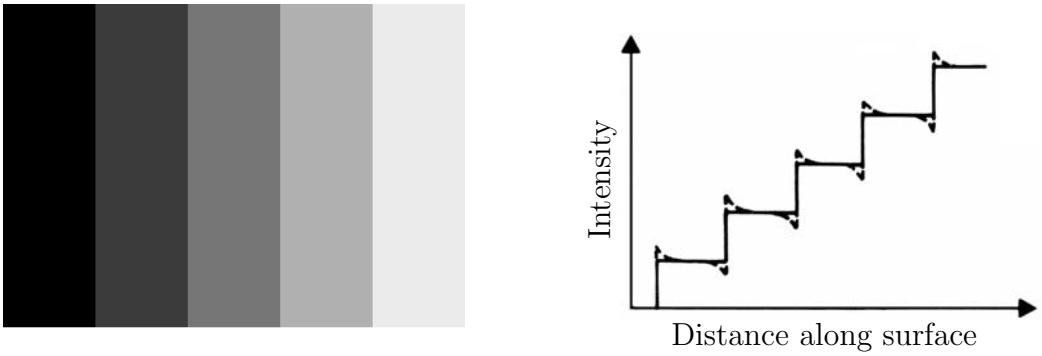
The range of light and dark that sensors in current digital cameras are able to capture is far less than the luminosity range that can be captured by the human eye. A typical digital representation of a high dynamic range scene will either record brighter areas as completely white, or darker areas as completely black, resulting in the permanent loss of a significant level of detail in the scene. High dynamic range techniques provide a software solution to this problem, by combining a set of images that capture correctly different parts of the same scene to produce an image that resembles more closely to the way in which the scene would be perceived by the human eye. Gradient domain high dynamic range compression algorithms are among the methods that enable rendering such images on current displays.

The field of view of standard current digital cameras is at its best comparable to that of the human eye. As a result, combining multiple parts of the same scene, possibly taken at different times, to produce a wider representation of the scene remains an active field of research due to its wide range of applicability. Radiographs, for example, are an important component of diagnosis. In dental medicine, for instance, panoramic radiographs are frequently used to examine developing teeth or to diagnose existing afflictions. The image acquisition process is done with X-ray beams and the exposure takes several seconds, in which the motion of the patient is restricted. An alternative solution to this imaging problem is to acquire several images at the same time from different angles. Then, these images can be stitched in the gradient domain and a panoramic radiograph can be obtained. In theory, this approach implies

shorter exposure time and consequently yields a more accurate representation due to less motion blur.

Current cinematographic productions are hard to imagine without special visual effects. In recent years, the quality of these visual effects has dramatically increased and computer generated scenes have become so realistic that the viewer cannot easily distinguish them from real life filmed scenes. A possible approach to this high end technology is to develop advanced image morphing techniques which combine gradient information contained in a series of images and allow a natural transformation of one digital image into another.

State of the art image processing algorithms are continuously proposed and developed targeting all these applications. The objective of these techniques is to combine and enhance digital scene representations yielding results with a greater level of detail than that offered by the best existing imaging devices or even by the human eye. A main reason for the need for such algorithms lies in the fundamental difference between the way in which image information is perceived by the human eye and the digital camera. It is known that human vision is sensitive to local contrasts, and less sensitive to absolute intensity values. This is known as the Mach band effect [15, 34], illustrated in Figure 1.1. The Mach band effect consists in a subconscious modification of the real gray level intensity near an edge of luminance discontinuity. Conversely, sensors in digital cameras are sensitive to absolute light changes and fail to capture



(a) Effect visualization **(b)** Graphical representation

Figure 1.1: The Mach band effect can be observed in the solid grayscale band sequence by noticing how the intensity close to the left of each band appears lighter than it is, whereas the intensity close to the right of each band appears darker than it is. In a graphical representation of the effect shown on the right-hand side, solid lines represent actual intensity and dashed lines represent perceived intensity.

local details at all illumination levels. This fundamental difference in perceiving light and color stirred researchers' interest, and led to the idea that the gradient of digital images may contain information which, controlled effectively, would produce more meaningful and visually pleasing image representations.

In image stitching, multiple parts of a scene are combined into a larger representation of the same scene, providing an enlarged view field and a greater level of detail. In image morphing, an image is changed into another through a seamless transition.

One way in which a seamless transition can be achieved from an image to the other is by a technique known in literature as the *optimal seam* [2,28]. Given two spatially aligned images, optimal seam techniques search in their common region for a curve with minimum intensity difference. Once such a curve is found, the composite image is constructed by pasting the image parts on the corresponding sides of the curve. If a curve where the difference in intensity between the two images is zero exists, the results produced by this approach are, indeed, optimal. In most practical applications, however, differences in the global luminosity of the images to be combined render optimal seam techniques as hopeless, because in this case such a curve does not exist.

A different approach to stitching and morphing algorithms is by blending the images to be combined in such a way that in the final mosaic the transition from one image to the other is imperceptible. Good blending techniques should produce seamless composite images by compensating for existing exposure or color differences between the component images and introduce as little distortion as possible in the overall result. Existing techniques of image blending operate on image intensities, on the wavelet representation of the image, or on the image gradient.

One of the first intensity based multiscale blending methods is presented in [10] and relies on pyramid decomposition. The intermediate images are decomposed into bandpass filtered components and the blending is done at each scale, in a transition zone inversely proportional to the spatial frequency content in the band. In a similar approach [26], pyramid decomposition is replaced with wavelet decomposition.

A well known intensity based blending technique is *feathering* [39]. In this method, the mosaic is a weighted average of all intermediate images. In the composite mosaic image, pixels are assigned weights proportional to their distance from the centre of the image they come from, resulting in a smoother transition from one image to the other in the final mosaic.

A technique called *Gradient-domain Image Stitching* is introduced by Levin et al. [30]. In their work, the authors suggest two gradient based approaches to image

stitching. One alternative is to obtain the composite image by minimizing a dissimilarity measure between the derivatives of the stitched image and the derivatives of the input images. The second approach is to simply blend the gradients of the component images in the overlap region, using existing intensity based blending techniques such as optimal seam, pyramid blending or feathering. The method proves that the gradient domain provides an attractive alternative to traditional blending methods. This very good performance of gradient based techniques over other methods resides in the fact that these algorithms operate on local contrast information, similarly to human vision.

1.2 Image reconstruction from gradient

Processing digital signals in the gradient domain has become subject of extensive research during the past decade. In the particular case of digital image processing, many algorithms [6,13,30,31] operating on gradient measurements have been proposed to date and proven successful, as an alternative or complement to the traditional intensity based methods.

A final step in gradient based techniques is to recover an estimate of the image from a given gradient data set, which may or may not be integrable. The problem of image reconstruction from gradient data is equivalent to finding a solution to the Poisson equation and various approaches to provide a solution have been proposed [5, 16, 24, 37].

The Frankot-Chellappa reconstruction algorithm [16] arose as a solution to the shape from shading problem in computer vision. The method relies on properties of the two-dimensional (2D) Fourier transform. Instead of reconstructing the image by integrating the gradient data in space domain, the Fourier transform of the gradient is computed and the image is obtained by computing the inverse Fourier transform of the frequency domain expression which corresponds to space domain integration. A discrete cosine transform (DCT)-based Poisson solver is presented by Simchony et al. [37]. In their work, the Poisson equation is solved in the DCT domain and the image is reconstructed by taking the inverse DCT of the solution.

The approach of Hampton et al. [24] is based on the observation that the Haar filters can be viewed as scaled discrete partial differentiation operators. An estimate of the Haar wavelet decomposition of the image to be reconstructed is obtained from measurements of discrete partial derivatives. An approximation of the image can be

obtained from this estimate of the Haar decomposition via Haar synthesis. Adding a Poisson solver during the synthesis stage improves the results, when the given gradient data set contains errors and/ or noise.

This reconstruction technique can be applied as a final step in any gradient based image processing application, such as high dynamic range imaging, panoramic imaging, image morphing or object insertion. In this thesis, this method will be applied to recover an image from a given gradient data set in image stitching and object insertion applications.

1.3 Thesis contribution and outline

The objective of this work is to study two applications of image processing, image stitching and object insertion, and to present new techniques for dealing with these applications in the gradient domain. For this purpose, first, a Haar wavelet based algorithm [23, 24] to recover an image from a given gradient is presented in detail, improved, and compared with two other reconstruction methods [16, 37]. Then, two gradient based algorithms for image stitching and object insertion are proposed. In these algorithms, the composite image is recovered from the gradient using the Haar wavelet based method.

The organization of this thesis is as follows:

In chapter 2, the problem of image reconstruction from gradient is introduced, motivated by the promising good performance of gradient based image processing algorithms. Several approaches to recover the image from a given gradient data set are reviewed. In particular, the image reconstruction technique proposed by Hampton et al. [23, 24] is presented in detail. The method is based on obtaining the Haar wavelet decomposition of the desired image directly from gradient measurements and then using Haar wavelet synthesis to reconstruct the image. This technique is further improved to produce better results, given a certain geometry of the gradient. A comparison is provided to illustrate with examples the performance of the analyzed reconstruction methods, under different conditions.

In chapter 3, two image processing problems and two methods are introduced for treating them. First, the problem of image stitching is presented and an algorithm is developed. The stitching algorithm has three main stages: wavelet based image registration, gradient based blending and image reconstruction from gradient. The stitching algorithm is gradually constructed to address situations of a gradually in-

creasing level of difficulty and a higher level of generality. Then, the problem of object insertion is presented in the context of image morphing. A conceptually simple object insertion method is described. The analysis performed in chapter 2 indicates that the reconstruction method of Hampton et al. can be employed as a final stage in both the stitching and the insertion techniques to recover the final composite image from the given gradient.

In chapter 4, the stitching and insertion algorithms proposed are applied to produce seamless composite images. The performance of the techniques is evaluated by analyzing the results obtained from a set of input images representative for several scenarios of increasing level of difficulty. In both analyzed applications, two criteria measure the quality of the composite image: similarity of the result to the input images, in the non-overlapping parts, and seamless, plausible transition from one input image to the other. The results presented in this chapter show that the techniques proposed in chapter 3 provide good solutions to the problems of image stitching and object insertion.

In chapter 5, the main ideas of this thesis are summarized and several directions for future research are suggested.

Chapter 2

Image reconstruction from gradient

2.1 Introduction

The fact that human visual system is more sensitive to local contrast changes than to global intensity values motivated gradient based algorithms in image editing [31], artifacts removal [6], high dynamic range compression [13] and panoramic imaging [2, 30, 36].

In section 2.2 of this chapter, various discretization models for the gradient are presented and the problem of image reconstruction from gradient is described. In section 2.3, several approaches to recover the image from a given gradient data set are reviewed. In particular, the method introduced by Hampton et al. in [23, 24] is described in detail. The method is further improved and shown to give very good results in the presence of moderate noise in the given data set. In section 2.4 the performance of several reconstruction methods is illustrated by examples and the quality of the results is evaluated under different conditions.

2.2 Preliminaries

2.2.1 Representing the image and its gradient

A grayscale image is represented analytically by a function of the form

$$\Phi : [1, M] \times [1, N] \to R$$

where M and N represent the width and height of the image, respectively. The set R is called the range of the image. The value $\Phi(x, y)$ is the pixel intensity value at location (x, y) in the image plane. The top left point of the image is often considered the origin of the coordinate system in the plane in which the image is represented.

A discrete representation views an image as a matrix Φ with M columns and N rows, in which each element $\Phi(x, y)$ represents the intensity value at location (x, y) in the image plane. This model is particularly useful in the implementation stage of image processing algorithms in programming languages such as MATLAB, in which matrix structures are efficiently stored and processed.

In the continuous case, the gradient of an image Φ is typically denoted by $\nabla\Phi$ and is given by:

$$\nabla\Phi(x_0, y_0) = \begin{bmatrix} \frac{\partial\Phi}{\partial x} \\ \frac{\partial\Phi}{\partial y} \end{bmatrix} (x_0, y_0) \tag{2.1}$$

In the discrete case, two approximation schemes [17, 27] are most commonly employed to compute the gradient of an image. The two discretization models are described in detail in what follows for reference.

The most frequently encountered gradient discretization model in image processing is the Hudgin [27] model. The Hudgin model is based on the forward difference scheme. Specifically, given a direction and a point (x, y), the value of the Hudgin derivative at that point in that direction is approximated by the difference between the intensity values of two neighbouring points, in that direction. That is, in the Hudgin geometry, assuming zero Neumann boundary conditions, the gradient $\tilde{\Phi}^H$ of an image Φ of size $M \times N$ is written as:

$$\tilde{\Phi}^H = \begin{bmatrix} \tilde{\Phi}^H_x \\ \tilde{\Phi}^H_y \end{bmatrix} \tag{2.2}$$

where the components $\tilde{\Phi}^H_x$ and $\tilde{\Phi}^H_y$ are matrices of size $M \times N$ and are given by:

$$\tilde{\Phi}^H_x(x, y) = \begin{cases} \Phi(x+1, y) - \Phi(x, y) & \text{for } 1 \le x \le M-1, 1 \le y \le N \\ 0 & \text{for } x = M, 1 \le y \le N \end{cases} \tag{2.3}$$

$$\tilde{\Phi}^H_y(x, y) = \begin{cases} \Phi(x, y+1) - \Phi(x, y) & \text{for } 1 \le x \le M, 1 \le y \le N-1 \\ 0 & \text{for } 1 \le x \le M, y = N \end{cases} \tag{2.4}$$

The superscript H in Equations 2.2-2.4 indicates the geometry employed in the discretization of the directional derivatives, namely the Hugdin geometry. The subscripts x and y indicate the direction in which the gradient component is computed.

Another discretization model employed to compute the directional derivatives of a 2D image is the Fried model [17]. The Fried model is frequently encountered in adaptive optics literature, due to the geometry of the Shack-Hartmann [47] wavefront sensors. In Fried geometry, the directional derivatives are computed in two steps:

- a "forward difference" operation in one direction:

$$\tilde{\Phi}_x^H(x,y) = \Phi(x+1,y) - \Phi(x,y), \text{ for } 1 \leq x \leq M-1, 1 \leq y \leq N \tag{2.5}$$

$$\tilde{\Phi}_y^H(x,y) = \Phi(x,y+1) - \Phi(x,y), \text{ for } 1 \leq x \leq M, 1 \leq y \leq N-1 \tag{2.6}$$

- an averaging operation, in the orthogonal direction:

$$\tilde{\Phi}_x^F(x,y) = \frac{\tilde{\Phi}_x^H(x,y) + \tilde{\Phi}_x^H(x,y+1)}{2}, \text{ for } 1 \leq x \leq M-1, 1 \leq y \leq N-1 \tag{2.7}$$

$$\tilde{\Phi}_y^F(x,y) = \frac{\tilde{\Phi}_y^H(x,y) + \tilde{\Phi}_y^H(x+1,y)}{2}, \text{ for } 1 \leq x \leq M-1, 1 \leq y \leq N-1 \tag{2.8}$$

We note that the first step in computing the Fried gradient produces, in fact, the Hudgin gradient of the image. Combining formulas 2.5-2.8, we obtain the expression of the Fried gradient:

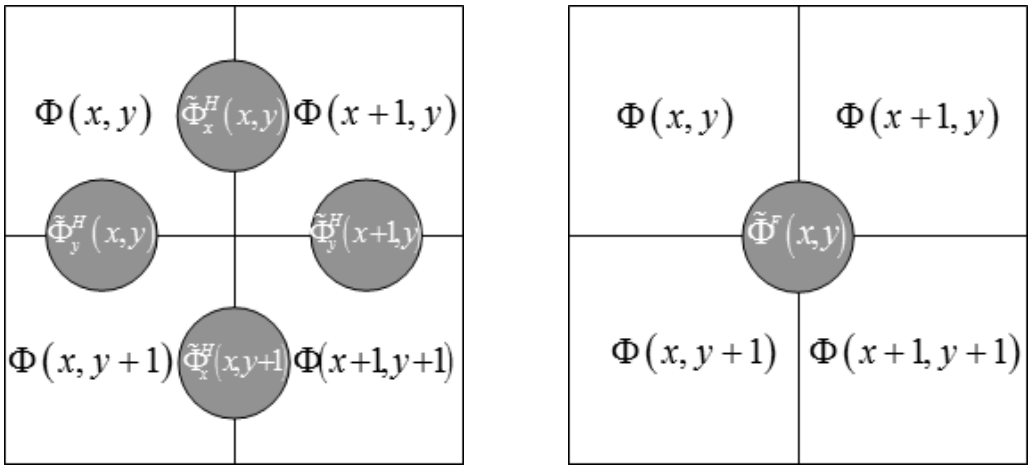
$$\tilde{\Phi}^F = \begin{bmatrix} \tilde{\Phi}_x^F \\ \tilde{\Phi}_y^F \end{bmatrix} \tag{2.9}$$

where the components $\tilde{\Phi}_x^F$ and $\tilde{\Phi}_y^F$ are matrices of dimension $(M-1)\times(N-1)$, with elements given by:

$$\begin{aligned} \tilde{\Phi}_x^F(x,y) &= \frac{1}{2}\left[\tilde{\Phi}_x^H(x,y) + \tilde{\Phi}_x^H(x,y+1)\right] \\ &= \frac{1}{2}\left[\Phi(x+1,y) - \Phi(x,y) + \Phi(x+1,y+1) - \Phi(x,y+1)\right] \end{aligned} \tag{2.10}$$

and

$$\begin{aligned} \tilde{\Phi}_y^F(x,y) &= \frac{1}{2}\left[\tilde{\Phi}_y^H(x,y) + \tilde{\Phi}_y^H(x+1,y)\right] \\ &= \frac{1}{2}\left[\Phi(x,y+1) - \Phi(x,y) + \Phi(x+1,y+1) - \Phi(x+1,y)\right] \end{aligned} \tag{2.11}$$



(a) Hudgin geometry **(b)** Fried geometry

Figure 2.1: Discretization models for computing image gradient. The gray circles are the values of the derivatives and the white squares are the original image intensity values.

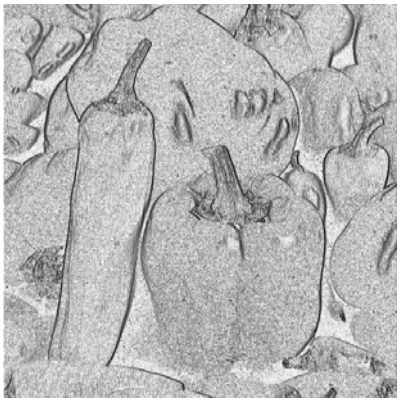
The superscript F in Equations 2.9-2.11 denotes the geometry employed in the discretization of the gradient, namely the Fried geometry.

The two discretization models are related by a simple averaging operation, as can be seen from Equations 2.10 and 2.11. Specifically, the value of a Fried directional derivative at a point simply represents the average of the values of two neighbouring Hudgin directional derivatives.

The Fried and Hudgin gradient approximation models are illustrated on a 2×2 pixel grid in Figure 2.1. The white squares represent intensity locations and values and the gray circles represent locations and values of the directional derivatives.

The significance of the information contained in the gradient becomes evident by analyzing Figure 2.2. On the top row, the gradient components of the standard test image `peppers` are shown. Small magnitude values of the gradient are graphically displayed in white, whereas high magnitude gradient values are shown in black. The magnitude of the gradient components can be understood by analyzing Figure 2.2c from the frequency content point of view. By comparing the image with its gradient components, it becomes clear that high magnitude gradient values correspond to abrupt light intensity changes in the actual image, or fine details, whereas low magnitude gradients indicate smooth transitions or flat patterns in the actual image.

The two gradient discretization models extend naturally to address the case of color images. In general, a color image is represented by a three-dimensional (3D)



(a) Fried gradient in x direction, depicts vertical edges.

(b) Fried gradient in y direction, depicts horizontal edges.

(c) Test image `peppers`

Figure 2.2: Top row: Gradient domain visualization. White indicates small intensity changes, black indicates significant intensity changes.

array, with two dimensions indicating the size of the picture (i.e., its height and width), and a third one indicating the number of channels used for the color representation. The most commonly employed color representation models are red green blue (RGB) and luminance, blue and red chrominance (YC_bC_r). In general, algorithms and further processing have to be performed on each channel of the color image. In particular, the gradient of a color image is obtained by treating each color channel as a grayscale image, for which the gradient can be obtained following one of the discretization models described. As a result, the discrete gradient of a color image is a two dimensional vector in which each component is a 3D array.

2.2.2 The problem of image reconstruction from gradient

Image editing, high dynamic range rendering or image stitching are thoroughly studied problems in image processing. Existing solutions to these problem can be classified with respect to the domain in which the processing is performed as intensity or gradient based. Many gradient based methods have been proposed to date [6, 13, 14, 30] and shown to yield good results, as an attractive alternative to algorithms which operate directly on pixel intensities or at various resolution scales [10, 42]. In gradient based image processing algorithms, image reconstruction from a given gradient data set is a required stage. As a result, developing robust reconstruction techniques has become subject to separate study and various methods have been proposed to date, addressing a number of circumstances [4, 5, 24].

Given an image Φ, computing its gradient is a simple, direct problem. The inverse and more challenging problem encountered in numerous image processing applications is how to recover an image from a given gradient data set.

In the continuous case, the problem of image reconstruction from a given gradient data set can be modelled by the following equation:

$$\nabla\Phi = \tilde{\Phi} \tag{2.12}$$

where the operator $\nabla = \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix}$ denotes the continuous gradient operator, Φ is the unknown image to be reconstructed, and $\tilde{\Phi} = \begin{bmatrix} \tilde{\Phi}_x \\ \tilde{\Phi}_y \end{bmatrix}$ is the given gradient data set.

Problem 2.12 may not always have a solution. It is known [13] that the gradient of a potential function is a *conservative vector field*. In the case of 2D functions such as images, the conservative field condition is equivalent to the zero curl condition and implies the equality of the second order mixed partial derivatives. Namely, Φ must satisfy the following condition:

$$\frac{\partial^2\Phi}{\partial x\partial y} = \frac{\partial^2\Phi}{\partial y\partial x} \tag{2.13}$$

Consequently, in order for problem 2.12 to have a solution, the vector field $\tilde{\Phi}$ must satisfy:

$$\frac{\partial\tilde{\Phi}_x}{\partial y} = \frac{\partial\tilde{\Phi}_y}{\partial x} \tag{2.14}$$

Unfortunately, this condition is rarely met in practice, because the given gradient data set $\tilde{\Phi}$ is either directly acquired from sensors, and therefore most likely contains measurement noise, or it is obtained from applying algorithms on an integrable gradient with the result that property 2.14 no longer holds.

A common approach to find an estimate of the image Φ is to reformulate Problem 2.12 as a minimization problem:

$$\min_{\Phi} \int\int \left[\left(\frac{\partial \Phi}{\partial x} - \tilde{\Phi}_x \right)^2 + \left(\frac{\partial \Phi}{\partial y} - \tilde{\Phi}_y \right)^2 \right] dxdy \tag{2.15}$$

A function Φ that minimizes 2.15, must satisfy the Euler-Lagrange equation:

$$2 \left(\frac{\partial^2 \Phi}{\partial x^2} - \frac{\partial \tilde{\Phi}_x}{\partial x} \right) + 2 \left(\frac{\partial^2 \Phi}{\partial y^2} - \frac{\partial \tilde{\Phi}_y}{\partial y} \right) = 0. \tag{2.16}$$

Dividing both sides of equation 2.16 by 2, yields:

$$\frac{\partial^2 \Phi}{\partial x^2} - \frac{\partial \tilde{\Phi}_x}{\partial x} = \frac{\partial \tilde{\Phi}_y}{\partial y} - \frac{\partial^2 \Phi}{\partial y^2}. \tag{2.17}$$

and

$$\frac{\partial^2 \Phi}{\partial x^2} + \frac{\partial^2 \Phi}{\partial y^2} = \frac{\partial \tilde{\Phi}_x}{\partial x} + \frac{\partial \tilde{\Phi}_y}{\partial y}. \tag{2.18}$$

A more compact form to write equation 2.18 is:

$$\nabla^2 \Phi = \nabla^T \cdot \tilde{\Phi} \tag{2.19}$$

where $\nabla^2 \Phi = \frac{\partial^2 \Phi}{\partial x^2} + \frac{\partial^2 \Phi}{\partial y^2}$ denotes the Laplacian of Φ, and the symbol $\cdot$ is used to denote the standard inner product.

Equation 2.19 is known as the Poisson equation and there are various approaches to solve it, given proper boundary conditions.

2.3 Approaches to image reconstruction from gradient

In image processing algorithms, modifications done in the gradient domain generally yield sets of data which are no longer conservative vector fields, and image reconstruc-

tion can no longer be attempted by direct integration. In adaptive optics, wavefronts are sensed by measuring the values of the derivatives, as opposed to measuring intensity values. In [41], a camera that captures derivative values of the imaged scene is proposed and shown to have great potential of applicability in digital photography. In both of these applications, the gradient data is measured using physical sensors. As it is known, measurement noise is inherent, and so reconstructing the image becomes again problematic. Despite the more challenging aspect of reconstructing the image in all of these situations, the techniques and devices described above seem to have a great potential of applicability (high dynamic range imaging, storage reduction, easier implementation of gradient based algorithms directly at camera level). As a result, the development of fast and robust reconstruction techniques has become subject to research interest in the past decades.

Undoubtedly, whether we are dealing with a gradient data set acquired directly from sensors, or with one obtained as a result of an algorithm, the main challenge that must be addressed by any reconstruction technique is robustness to the potentially non-conservative nature of the gradient vector field to be integrated. In [16] and [37] two reconstruction methods are proposed, addressing specifically the case of image reconstruction from non-integrable gradient data sets. The problem is also largely investigated in [5] and in [33].

In [24], a 2D numerical integration technique is proposed. The method is based on obtaining the Haar wavelet decomposition of the image to be reconstructed directly from the gradient via algebraic manipulations, hence its performance is influenced by underlying errors in the given gradient. To overcome this problem and yield acceptable results even in the presence of high magnitude noise, the method can be improved by incorporating a Poisson solver during the reconstruction [23].

In section 2.3.1 a brief overview of two reconstruction methods in [16, 37] is presented and in section 2.3.2 the reconstruction method in [24] is described in detail. It is shown how the results produced by this method can be further improved, in two different scenarios. The first approach leading to better results, introduced in this thesis, implies using information from the Hudgin gradient, as is common in image processing and computer vision application. This method results in almost perfect reconstruction in the case of moderate errors in the input gradient. The second improvement was introduced in [23] and addresses specifically the case of more noise in the given gradient.

2.3.1 Fourier and discrete cosine transform image reconstruction

In [16] a surface reconstruction method is proposed as a possible approach to the computer vision problem of shape from shading. The method is often referred to as the Frankot-Chellappa algorithm and relies on properties of the 2D Fourier transform. Specifically, in this technique, instead of reconstructing the image by integrating the gradient data in the space domain, the 2D discrete Fourier transform (DFT) of the two gradient components is computed and an estimate of the image is obtained by taking the inverse Fourier transform of a linear combination thereof.

In [37], a DCT-based 2D Poisson solver is presented. The Poisson equation is solved in the DCT domain and the image is reconstructed by taking the inverse DCT of the result obtained.

2.3.2 Haar wavelet based reconstruction algorithm

Initial framework and formulation

A fast wavelet based image reconstruction method is proposed by Hampton et al. in [24]. The setup is the following: given a Fried aligned gradient data with components of size $\left(2^M - 1\right) \times \left(2^M - 1\right)$, the objective is to integrate it and obtain an image.

The algorithm is defined by two main stages: first, the Fried aligned gradient (see section 2.2.1 for details) is reorganized into an estimate of the Haar wavelet decomposition of the desired image, and then the image is reconstructed from this decomposition via Haar wavelet synthesis. In Figure 2.3 a diagram of the reconstruction method is shown.

This approach is mainly motivated by the practical necessity to reconstruct wavefronts from sensed gradient measurements in adaptive optics. In such applications, the sensors are known to capture derivatives, and the physical alignment of the sensors is consistent with the Fried model.

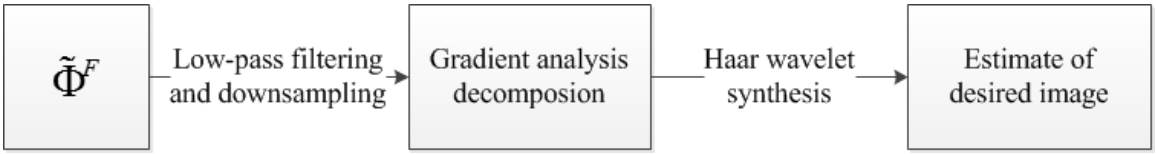
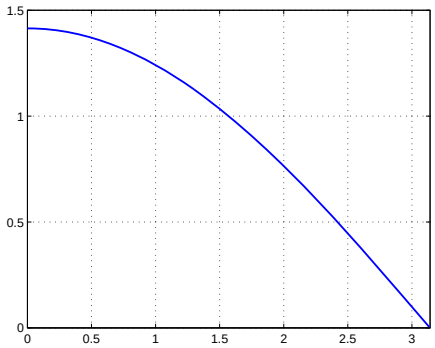
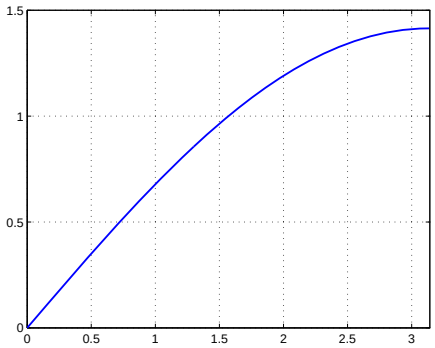


Figure 2.3: Hampton et al. reconstruction algorithm



(a) Amplitude response of $H_L(z)$



(b) Amplitude response of $H_H(z)$

Figure 2.4: Amplitude response of Haar analysis filters

Computing the Fried gradient can be regarded as a succession of one-dimensional (1D) filtering operations, performed along consecutive rows and columns of an image. The key idea of the reconstruction method proposed in [24] is the relationship between the filters that model the directional derivatives in Fried alignment, and the filters known in literature as the Haar filters. In what follows, this relationship is explained in detail.

The Haar analysis filters are given by:

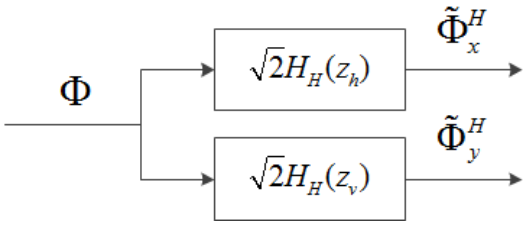
$$H_L(z) = \frac{1}{\sqrt{2}}\left(1 + z^{-1}\right) \tag{2.20}$$

$$H_H(z) = \frac{1}{\sqrt{2}}\left(1 - z^{-1}\right) \tag{2.21}$$

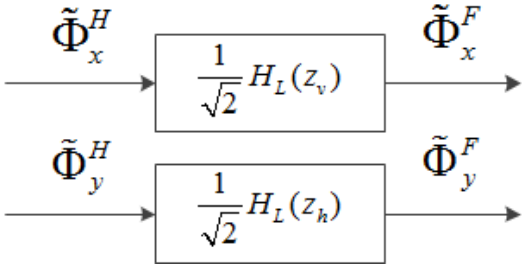
The behaviour of the Haar analysis filters is illustrated by their amplitude response plots shown in Figure 2.4. In particular, it is clear that $H_L(z)$ is a scaled averaging linear phase low-pass filter, whereas $H_H(z)$ is a scaled differencing linear phase high-pass filter.

Computing the discrete gradient of the image in either of the geometries previously described in section 2.2.1 can be done by filtering the image with scaled versions of the Haar analysis filters.

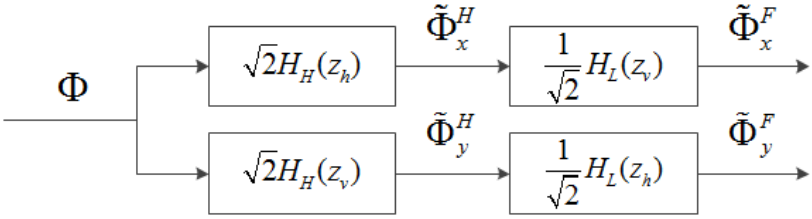
To understand how the Hudgin gradient components can be obtained from the image via filtering, with the definitions 2.2-2.4 in mind, consider the definition of the highpass analysis Haar filter. It is clear that the Hudgin gradient $\tilde{\Phi}^H$ can be obtained from the original image Φ by filtering the image with a scaled version of



(a) Hudgin gradient obtained from image via scaled Haar filtering



(b) Fried gradient obtained from Hudgin gradient via scaled Haar filtering



(c) Fried gradient obtained by successive scaled Haar filtering

Figure 2.5: Gradients computed via scaled Haar filters

the Haar analysis highpass filter (the scaling factor is necessary in order to preserve the magnitude of the gradient). This is illustrated diagrammatically in Figure 2.5a, where the right-hand side subscripts h and v in z_h and z_v indicate the horizontal, and vertical direction in which the filtering is performed, respectively. Pursuing this further, the Fried gradient is obtained by filtering each Hudgin gradient component with a scaled version of the lowpass analysis Haar filter.

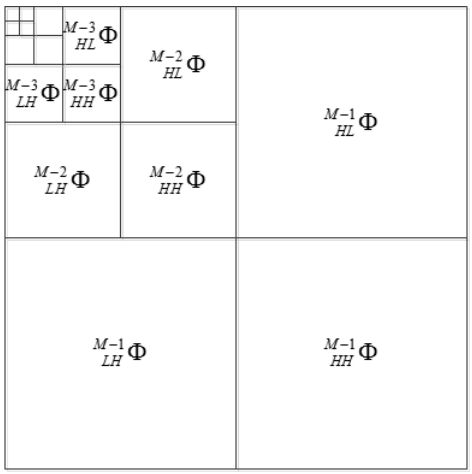


Figure 2.6: Standard Haar decomposition

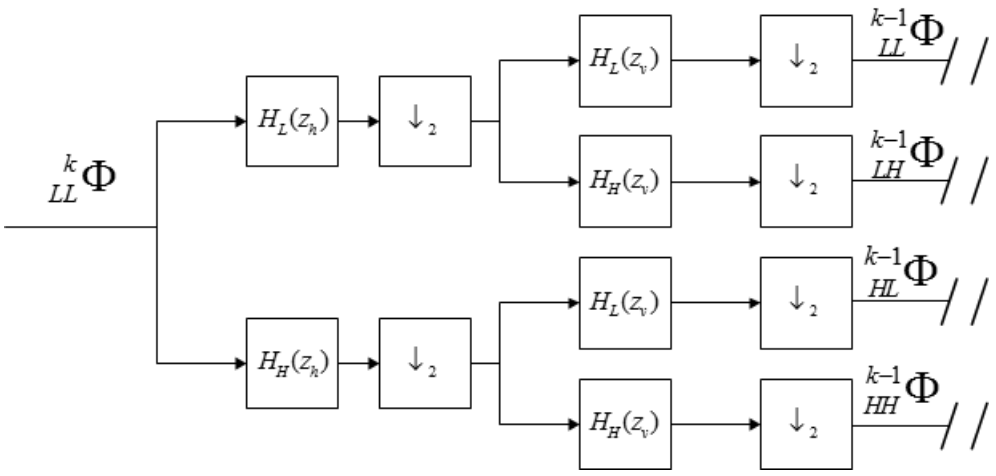


Figure 2.7: Obtaining one level in a Haar wavelet decomposition

From wavelet theory [44], it is known that an image Φ of size $2^M \times 2^M$ can be reorganized as an M-level structure, known as the wavelet decomposition. In a wavelet decomposition, the first level is obtained by filtering the image with the analysis filters, followed by a downsampling operation, necessary in order to preserve the dimensionality of the structure. An M-level Haar wavelet decomposition is shown in Figure 2.6. A 1-level wavelet decomposition has four main frequency bands (quadrants), indicated by the lower left subscripts HH, LL, HL and LH. The significance of these subscripts is given by the order in which the Haar wavelet filters are applied to the image. The filtering and downsampling operations are performed on the LL quadrant to obtain the next level of the decomposition.

In Figure 2.7, obtaining level $k-1$ of a wavelet decomposition from the previous level k is illustrated. The symbol $\downarrow$ denotes downsampling, by the factor indicated on its right-hand side. The significance of the left-hand side superscript k is twofold. On the one hand, k indicates the level in the wavelet decomposition. On the other hand, k provides information on the size of $^{k}_{LL}\Phi$, specifically $^{k}_{LL}\Phi$ is of size $2^k \times 2^k$.

According to [24] and as illustrated in Figure 2.5, the relationship between the Fried aligned gradient of an image and the Haar analysis filters is described by:

$$^{M}\tilde{\Phi}_x = {}^{M}\Phi H_H(z_h)H_L(z_v) \; ^{M}\tilde{\Phi}_y = {}^{M}\Phi H_L(z_v)H_H(z_v) \tag{2.22}$$

where the upper left superscript M indicates that the size of full resolution image $^{M}\Phi$ is $2^M \times 2^M$ and $^{M}\tilde{\Phi}_x$ and $^{M}\tilde{\Phi}_y$ are the two Fried gradient components of the full resolution image $^{M}\Phi$.

Based on this observation, the given gradient data is filtered and downsampled to obtain the M-level structure shown in Figure 2.8.

The equations to obtain this configuration are introduced in [24] and listed in what follows.

For the upper right and lower left quadrants (HL and LH, respectively):

$$
\begin{aligned}
{}^{M-1}_{HL}\tilde{\Phi} &= \downarrow_2 \left\{ {}^{M}\Phi H_H\left(z_h\right) H_L\left(z_v\right)\right\} = \downarrow_2 \left\{ {}^{M}\tilde{\Phi}_x\right\} && (2.23)\\
{}^{M-1}_{LH}\tilde{\Phi} &= \downarrow_2 \left\{ {}^{M}\Phi H_L\left(z_h\right) H_H\left(z_v\right)\right\} = \downarrow_2 \left\{ {}^{M}\tilde{\Phi}_y\right\} && (2.24)
\end{aligned}
$$

For the upper left quadrant (LL):

For $k = M$ to 2

$$
\begin{aligned}
{}^{k-1}\tilde{\Phi}_x &= \sqrt{2} \downarrow_2 \left\{ {}^{k}\tilde{\Phi}_x H_L^2\left(z_h\right) H_L\left(z_v^2\right)\right\} && (2.25)\\
{}^{k-1}\tilde{\Phi}_y &= \sqrt{2} \downarrow_2 \left\{ {}^{k}\tilde{\Phi}_y H_L\left(z_h^2\right) H_L^2\left(z_v\right)\right\} && (2.26)\\
{}^{k-2}_{HL}\tilde{\Phi} &= \downarrow_2 \left\{ {}^{k-1}\tilde{\Phi}_x\right\} && (2.27)\\
{}^{k-2}_{LH}\tilde{\Phi} &= \downarrow_2 \left\{ {}^{k-1}\tilde{\Phi}_y\right\} && (2.28)\\
{}^{k-2}_{HH}\tilde{\Phi} &= \sqrt{2} \downarrow_4 \left\{ {}^{k}\tilde{\Phi}_x H_L^2\left(z_h\right) H_H\left(z_v^2\right)\right\} \\
&= \sqrt{2} \downarrow_4 \left\{ {}^{k}\tilde{\Phi}_y H_H\left(z_h^2\right) H_L^2\left(z_v\right)\right\} && (2.29)
\end{aligned}
$$

For the lower right quadrant (HH):

For $k = M$ to 2

$$
{}^{k-1}\hat{\Phi}_x = \begin{cases} \sqrt{2} \downarrow_2 \left\{ {}^{k}\tilde{\Phi}_x H_L\left(z_h^2\right) H_H^2\left(z_v\right)\right\} & \text{for } k = M \\ \sqrt{2} \downarrow_2 \left\{ {}^{k}\hat{\Phi}_x H_L\left(z_h^2\right) H_L^2\left(z_v\right)\right\} & \text{for } k < M \end{cases} \quad (2.30)
$$

$$
{}^{k-1}\hat{\Phi}_y = \begin{cases} \sqrt{2} \downarrow_2 \left\{ {}^{k}\tilde{\Phi}_y H_H^2\left(z_h\right) H_L\left(z_v^2\right)\right\} & \text{for } k = M \\ \sqrt{2} \downarrow_2 \left\{ {}^{k}\hat{\Phi}_y H_L^2\left(z_v\right) H_L\left(z_v^2\right)\right\} & \text{for } k < M \end{cases} \quad (2.31)
$$

$$
{}^{k-2}_{LH}\hat{\Phi} = \downarrow_2 \left\{ {}^{k-1}\hat{\Phi}_x\right\} \quad (2.32)
$$

$$
{}^{k-2}_{HL}\hat{\Phi} = \downarrow_2 \left\{ {}^{k-1}\hat{\Phi}_y\right\} \quad (2.33)
$$

$$
\begin{aligned}
{}^{k-2}_{HH}\hat{\Phi} &= \begin{cases} \sqrt{2}\downarrow_4 \{{}^{k}\tilde{\Phi}_x H_H\left(z_h^2\right) H_H^2\left(z_v\right)\} & \text{for } k = M \\ \sqrt{2}\downarrow_4 \{{}^{k}\hat{\Phi}_x H_H\left(z_h^2\right) H_L^2\left(z_v\right)\} & \text{for } k < M \end{cases} \\
&= \begin{cases} \sqrt{2}\downarrow_4 \{{}^{k}\tilde{\Phi}_y H_H^2\left(z_h\right) H_H\left(z_v^2\right)\} & \text{for } k = M \\ \sqrt{2}\downarrow_4 \{{}^{k}\hat{\Phi}_y H_L^2\left(z_h\right) H_H\left(z_v^2\right)\} & \text{for } k < M \end{cases}
\end{aligned} \tag{2.34}
$$

If we iterate Equations 2.25 - 2.34 for each $k = M$ to 2, the final gradient decomposition will be missing two quadrants: ${}^{0}_{LL}\tilde{\Phi}$ (the top left value in the top left half resolution quadrant of the decomposition shown in Figure 2.8) and ${}^{0}_{LL}\hat{\Phi}$ (the top left value in the bottom right half resolution quadrant of the decomposition shown in Figure 2.8). The value of ${}^{0}_{LL}\Phi$ is proportional to the average intensity of the desired image and can be either measured by an additional sensor, or estimated from the dynamic range of the image to be reconstructed. As is noted in [24], the other missing value, ${}^{0}_{LL}\hat{\Phi}$, can be set to zero and ignored without significant consequences in the reconstruction. Another approach for identifying this value will be discussed later in this section.

Given the gradient analysis decomposition in Figure 2.8, the image is reconstructed in two stages. First, an $(M-1)$-level Haar wavelet synthesis step is applied to quadrant HH in Figure 2.8, resulting in the Haar decomposition of the desired image. Then, the image is obtained from this Haar decomposition, after an M-level Haar wavelet synthesis step.

Haar wavelet synthesis means upsampling by two, followed by row and column-wise filtering with the Haar synthesis filters.

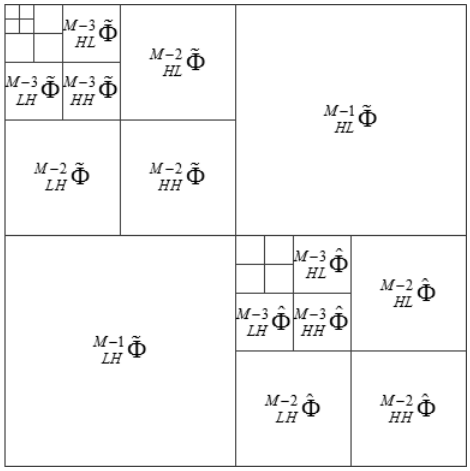


Figure 2.8: Gradient analysis decomposition

To ensure shift free perfect reconstruction, the synthesis filters are described by:

$$G_L(z) = H_L\left(z^{-1}\right) = \frac{1}{\sqrt{2}}(1+z)\, G_H(z) = H_H\left(z^{-1}\right) = \frac{1}{\sqrt{2}}(1-z) \tag{2.35}$$

In these notations, obtaining level $k+1$ from level k can be done by:

$$\begin{aligned} {}^{k+1}_{LL}\Phi =& G_L(z_h)\left(G_L(z_v)\uparrow_2 \{{}^{k}_{LL}\Phi\} + G_H(z_v)\uparrow_2 \{{}^{k}_{LH}\Phi\}\right) \\ &+ G_H(z_h)\left(G_L(z_v)\uparrow_2 \{{}^{k}_{HL}\Phi\} + G_H(z_v)\uparrow_2 \{{}^{k}_{HH}\Phi\}\right) \end{aligned} \tag{2.36}$$

where the symbol $\uparrow$ denotes upsampling by the factor indicated by the right-hand side subscript, and $k = \{0, 1, \ldots, M-1\}$.

It is important to note that the discretization of the given gradient does not restrict the applicability of the algorithm. Considering the simple method to convert Hudgin aligned gradient to Fried aligned gradient (see section 2.2.1 for details), it is clear that the image reconstruction algorithm can easily be extended to accept this type of gradient as input. Moreover, using Hudgin aligned gradient as input allows for perfect reconstruction in the presence of no noise, given the mean value of the image to be reconstructed, as will be discussed in what follows and demonstrated by experiments in section 2.4.1. As a result, the method (in any of its versions which will shortly be described in this section), can be used to reconstruct the image at the end of any existing gradient domain based image processing algorithm, provided that the discretization of the gradient follows one of the two models described in section 2.2.1.

Extensions of the Haar wavelet based reconstruction algorithm

A. The mean value of the image

A first observation concerns the mean value of the image to be reconstructed, in the general rectangular shape case. In the original form of the algorithm, as proposed in [24] and described earlier in this section for reference, the mean value of the reconstruction can be corrected either before or after the synthesis step. Correcting before the synthesis is done by:

$${}^{0}_{LL}\Phi = 2^M \cdot m \tag{2.37}$$

where m is an estimate for the mean value of the image to be reconstructed.

Because the decomposition obtained after the analysis step is an estimate of the wavelet decomposition of the image to be reconstructed, formula 2.37 is only adequate if the image to be reconstructed is of size $2^M \times 2^M$. When the image to be reconstructed is not $2^M \times 2^M$, the value of ${}^0_{LL}\Phi$ in the decomposition would be proportional to the mean value of the reconstruction, but this reconstruction is not the image of interest, as can be seen from Figure 2.10. So, for a better estimate of the image, in this more general setup, the value of ${}^0_{LL}\Phi$ is set to zero and the mean value of the reconstruction is corrected after the synthesis step, only for the image of interest. If the mean value of the image to be reconstructed is unknown in practice, a reasonable estimate can be obtained from its dynamic range.

B. Correcting the waffle pattern

In this thesis, the reconstruction method proposed by Hampton et al. [24] is improved using information contained in the given Hugdin gradient. In the presence of no noise in the input gradient, the reconstruction error consists of a checkerboard pattern, known in adaptive optics literature as the waffle pattern. The waffle pattern in the reconstruction error is generated by the fact that, due to the Fried alignment of the gradient data, the information in ${}^0_{LL}\hat{\Phi}$ in the decomposition displayed in 2.8 is lost. In some applications, the information in ${}^0_{LL}\hat{\Phi}$ is not important and can be ignored without significant consequences in the reconstruction. However, if the application at hand requires a higher degree of reconstruction accuracy, the information contained in ${}^0_{LL}\hat{\Phi}$ can be recovered from the Hudgin gradient of the image to be reconstructed.

To understand how the algorithm of [24] was modified in this thesis to produce a higher degree of solution accuracy, consider the diagram in Figure 2.9. The given

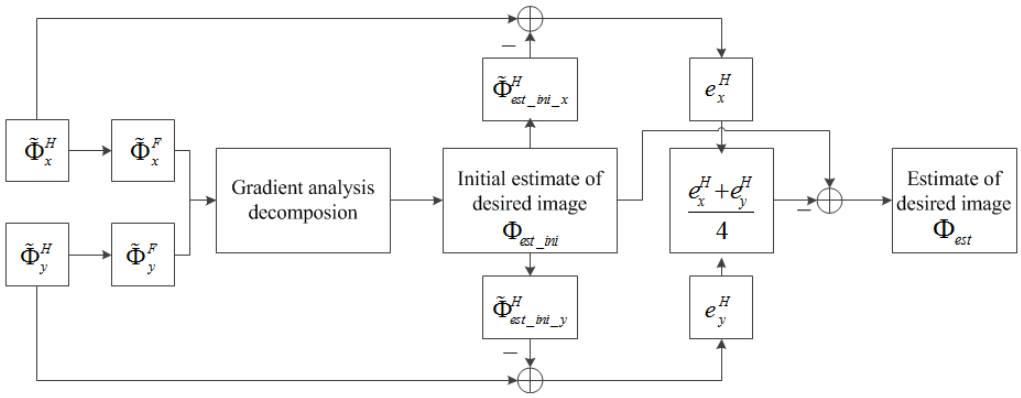


Figure 2.9: Reconstruction algorithm modified to correct waffle pattern error

Hudgin aligned gradient data set $\tilde{\Phi}^H$ is first converted to Fried aligned gradient $\tilde{\Phi}^F$. Next, lowpass filtering and downsampling operations are applied to the Fried aligned gradient to obtain the decomposition in Figure 2.8. Then, an initial estimate of the desired image Φ_{est_ini} is obtained via synthesis, using also the mean value of the desired image. Next, the Hudgin gradient $\tilde{\Phi}^H_{est\_ini}$ of this initial estimate is computed and subtracted from the given Hudgin gradient $\tilde{\Phi}^H$. The x and y components of this difference have the same 2×2 pattern. This pattern represents a scaled version of the difference between the initial estimate of the reconstruction and the original image (ground truth). For an enhanced result of the algorithm, this scaled difference term is subtracted from the initial image estimate, yielding an extremely accurate recovery of the ground truth image in the presence of almost no noise in the input gradient, as will be illustrated in the next section.

In summary, the modification of the algorithm of [24] proposed in this thesis can be attempted when the given gradient is in Hudgin alignment, as is often the case in image processing applications. The reconstructions produced by the modified technique have a high degree of accuracy in the presence of almost no noise in the input gradient as will be illustrated in section 2.4.

C. Obtaining an image from non-conservative vector fields

The fact that the reconstruction method relies exclusively on algebraic manipulation of the input directional derivatives, produces undesirable artifacts in the reconstructed image in the presence of noise in the input gradient. To overcome this problem, in [23], the method is improved by employing an iterative Poisson solver at each resolution during the synthesis. Specifically, the image estimate obtained at each resolution level during the synthesis stage is smoothened using a Poisson solver and used to reconstruct the next level.

In section 2.2.2, the problem of image reconstruction from the gradient was analytically described as the solution of the Poisson equation:

$$\nabla^2 \Phi_{est} = \nabla^T \cdot \tilde{\Phi} \tag{2.38}$$

where Φ_{est} is the image we want to recover from the given gradient $\tilde{\Phi}$. If we regard computing the Laplacian and gradient as convolutions, in the Fried discretization

model, Equation 2.38 becomes:

$$\begin{bmatrix} 1 & 0 & 1 \\ 0 & -4 & 0 \\ 1 & 0 & 1 \end{bmatrix} \otimes \Phi_{est} = \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix} \otimes \tilde{\Phi}_x + \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix} \otimes \tilde{\Phi}_y \tag{2.39}$$

As has been noted in [23], Equation 2.39 can be solved iteratively, using the Jacobi method. The resulting recursive formula is:

$$\Phi_{est}[i+1] = \Phi_{est}[i] - \frac{1}{4}\left(\begin{bmatrix} -1 & 0 & -1 \\ 0 & 4 & 0 \\ -1 & 0 & -1 \end{bmatrix} \otimes \Phi_{est}[k] + \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix} \otimes \tilde{\Phi}_x[i] + \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix} \otimes \tilde{\Phi}_y[i] \right) \tag{2.40}$$

where i is the iteration index.

At each resolution level, the LL quadrant obtained in the previous iteration is used as an initial point in Equation 2.40. The fact that a good initial point is provided at each resolution level ensures a fast convergence. In fact, depending on the level of noise present in the input gradient (measured by the signal to noise ratio (SNR)), a single iteration of Equation 2.40 at each resolution level yields acceptable results, as will be demonstrated in section 2.4. Performing one iteration of Equation 2.40 at each resolution level corresponds in the synthesis stage of the reconstruction algorithm to replacing each ${}^{k+1}_{LL}\Phi$ with:

$$\begin{aligned} {}^{k+1}_{LL}\Phi = & \frac{1}{2}\{ {}^{k+1}_{LL}\Phi z_h z_v H_L\left(z_h^2\right) H_L\left(z_v^2\right) - \\ & - {}^{k+1}\tilde{\Phi}_x H_H(z_h) H_L(z_v) - {}^{k+1}\tilde{\Phi}_y H_L(z_h) H_H(z_v)\} \end{aligned} \tag{2.41}$$

D. Reconstructing rectangular images of arbitrary size In the initial framework and formulation presented earlier in this section, the image to be reconstructed was assumed to be square, of size $2^M \times 2^M$. Images of various sizes are more likely to be encountered in practice. Consequently, modifications must be made to the algorithm if the image to be reconstructed is rectangular, with a size different than $2^M \times 2^M$.

To address this situation, a modification of the input gradient must be performed before the gradient analysis stage. As it is known, the 2D Haar wavelet transform applied to a $2^M \times 2^M$ image yields a decomposition of the same size. One approach, proposed in [25], when the image to be reconstructed is of a different size than $2^M \times$

2^M is to reflect (mirror) the input gradient and create a data set with two square components of size one unit smaller than the nearest power of two. Then, the Haar wavelet decomposition of this modified gradient data set is obtained and the image is reconstructed via synthesis. The effect of reflecting gradient data as described above has on the reconstruction is illustrated in Figure 2.10.

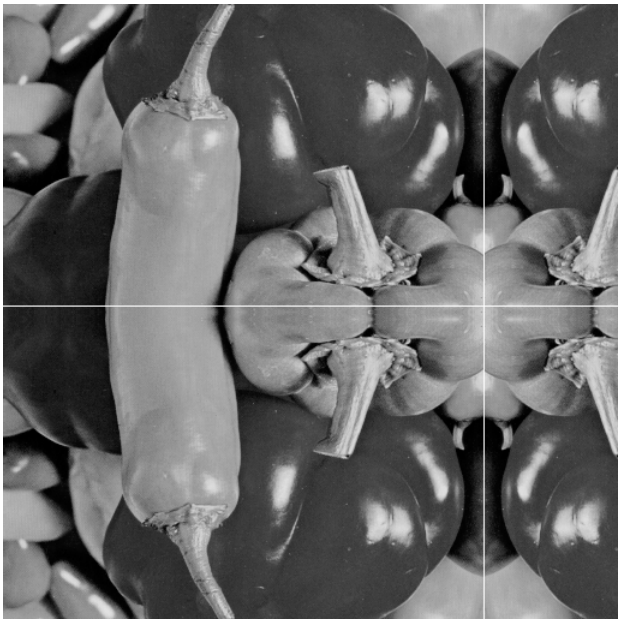


Figure 2.10: Image reconstructed from a reflected gradient. The image of interest is in the top left corner. The white lines represent the reflection axes.

The size of the image of interest is deduced from the size of the input gradient and from the discretization model considered in computing the directional derivatives. For example, a gradient with two components of size 250×399 in Fried geometry corresponds to an image of size 251×400. The decomposition obtained after the gradient analysis stage as described above and in section 2.3.2 will be of size 512×512. This yields a reconstruction of the same size (shown in Figure 2.10), where the image of interest is placed in the top left part, and has a size of 251×400. So, the image of interest can be cropped from the reconstruction obtained, given its size and location.

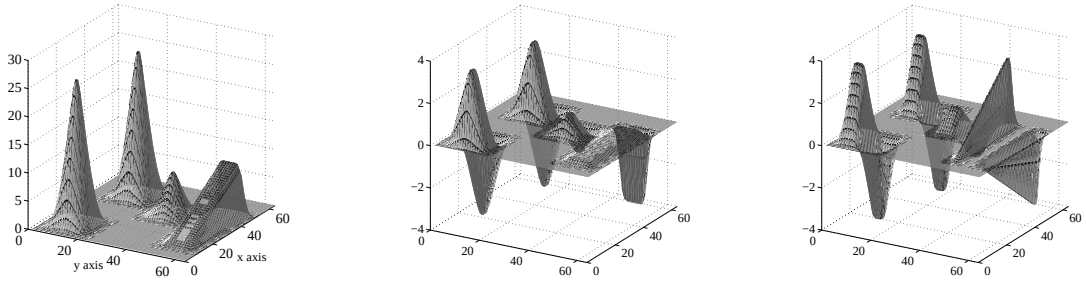
2.4 Comparison of image reconstruction methods

Given a gradient data set and the mean value of an image to be reconstructed, the objective is to obtain an estimate of the image. In this section, the performance of four reconstruction techniques is evaluated and compared by examples. The four algorithms are tested on images of various resolutions, and the quality of the reconstructions obtained is compared. The two main performance criteria analyzed are the accuracy of the solution produced and the computational speed of the method, in terms of computational complexity and computation time. First, the case of image reconstruction from noise free input gradient is studied. Then, the case of reconstructing the image from a gradient corrupted by white Gaussian noise is investigated.

The first analyzed reconstruction method is the fast DCT-based Poisson solver proposed in [37]. The second reconstruction method is the DFT-based Frankot-Chellappa algorithm [16]. The third reconstruction method is the Haar based wavelet integration method of Hampton et al. [24], modified as proposed in the previous section to correct the waffle pattern error. In the presence of no noise in the input gradient, this method is shown to yield the most accurate results. The fourth reconstruction method is the method of Hampton et al. with the addition of a Poisson solver during the synthesis stage [23].

2.4.1 Image reconstruction from a clean gradient data set

The compared algorithms are tested to reconstruct the images shown in Figure 2.11a and in Figure 2.12. The resolution of the test images is between 64×64 and 1024×1024.



(a) Test image **(b)** Hudgin x gradient **(c)** Hudgin y gradient

Figure 2.11: Test image `ramp peaks`: computer generated surface. Image size: 64×64.

(a) Test image `airplane`

(b) Test image `camera`

(c) Test image `barbara`

(d) Test image `man`

Figure 2.12: Standard test images of various sizes used in the comparison

The image `ramp peaks` was chosen as benchmark by Agrawal et al. in [5,33] and used therein in the context of surface reconstruction to compare the performance of the DCT-based Poisson solver and the DFT-based Frankot-Chellappa technique with other reconstruction methods[1]. The image represents a computer generated surface displayed as a three dimensional plot in which the vertical axis indicates pixel intensity values. The range of the image is [0..26] and the size of the image is 64×64. The final

[1]Source of MATLAB code: `http://www.umiacs.umd.edu/~aagrawal/software.html`

step of generating the original surface is low-pass filtering for a smoothened visual aspect. As a result of this, in the gradient domain, all magnitudes are significantly reduced. To enhance visualization of the analyzed surface, in Figure 2.11, the two components of the Hudgin aligned gradient (assuming zero Neumann conditions) are displayed.

The images shown in Figure 2.12 are standard test images with intensity values in the range [0..255]. The resolutions of these test images cover a range from 128×128 to 1024×1024. It should be specified that the analyzed reconstruction techniques do not require a specific size of the input gradient. That is, the input gradient can be of arbitrary size, as long as it is consistent with the image and follows the Hudgin discretization model.

Before the results are presented, the framework in which the experiments were conducted and details concerning the implementation of the algorithms will be provided.

An identical setting was ensured as starting point to allow a fair comparison of the algorithms studied. In this sense, first, the input gradient data set for all reconstruction algorithms was organized in Hudgin alignment (see section 2.2.1 for details). Second, the mean value of the image to be reconstructed was considered known and used to correct all reconstructions. As discussed in section 2.3.2, correcting the mean value of the reconstruction obtained by the method of Hampton et al. [24] can be done in two ways. Specifically, the mean value of the image can be adjusted either before the synthesis stage, by correcting the top left value of the gradient decomposition obtained, or after the synthesis stage, by uniformly shifting the amplitude of the reconstruction to obtain the desired average value. For all analyzed algorithms, the mean value of the reconstruction produced was corrected after the reconstruction.

The accuracy of the results produced by the studied algorithms was measured by computing the *relative error* of the reconstruction, given by:

$$re = \frac{||\Phi - \Phi_{est}||_F}{||\Phi||_F} \tag{2.42}$$

where Φ represents the original image (ground truth), Φ_{est} represents the reconstructed image, and $\|\cdot\|_F$ denotes the Frobenius norm (i.e., the square root of the sum of the squares of all elements in a matrix).

The relative errors of the image reconstructed by each algorithm for the five analyzed images are displayed in Table 2.1. It was observed that the performance of the

Table 2.1: Relative error of reconstructions in the case of no noise in the input gradient

Image	DCT Poisson solver [37]	Frankot-Chellappa [16]	Method from [24]	Method from [23]
ramp peaks	2.9784×10^{-15}	0.1459	1.2634×10^{-12}	0.0081
airplane	5.7518×10^{-15}	0.1015	0	0.0081
camera	1.0890×10^{-13}	0.1841	0	0.0136
barbara	2.1632×10^{-13}	0.1490	0	0.0129
man	7.3948×10^{-13}	0.2012	0	0.0108

algorithms is consistent in behaviour for all analyzed images. Specifically, the reconstruction method of Hampton et al. [24], with the correction for the waffle pattern as proposed in the previous section, yields the most accurate results and this is to be expected when the test images are in 8-bit representation and contain no noise. The method is followed closely by the DCT based Poisson solver in [37]. The algorithm of Hampton et. al [23] with the Poisson solver included at all resolutions during synthesis places the third, and the DFT-based Frankot-Chellappa algorithm [16] ranks the fourth.

In terms of computational complexity, the quality of the results produced by the studied algorithms was measured by computing the CPU time per reconstruction. A 64 bit Windows 7 Laptop PC with an Intel® Core ™ i5-430M CPU @ 2.27GHz, with 4GB DDR3 @ 1066 MHz of RAM equipped with MATLAB R2011b was used to run the codes and the average reconstruction times (in seconds) are shown in Table 2.2.

Another representation of the computational performance of the analyzed algorithms in terms of speed is shown in Figure 2.13. In this figure, the reconstruction times shown in Table 2.2 are represented versus image size. It can be observed that for low image size, such as 64×64, the computational performance of the algorithms is similar. For image sizes higher than 128×128, the DFT-based method [16] performed the best, followed by the reconstruction method of Hampton et al. [24], containing the correction for the waffle pattern proposed in the previous section.

Several remarks are in order to allow a better understanding of the speed performance of the compared algorithms. The results presented in Table 2.2 and in Figure

Table 2.2: Reconstruction CPU time (in seconds)

Image	DCT Poisson solver [37]	Frankot-Chellappa [16]	Method from [24]	Method from [23]
ramp peaks	0.0025	0.0008	0.0028	0.0039
airplane	0.0073	0.0034	0.0054	0.0081
camera	0.0309	0.0163	0.0193	0.0342
barbara	0.2347	0.0902	0.0992	0.1824
man	1.1153	0.3980	0.4911	0.8599

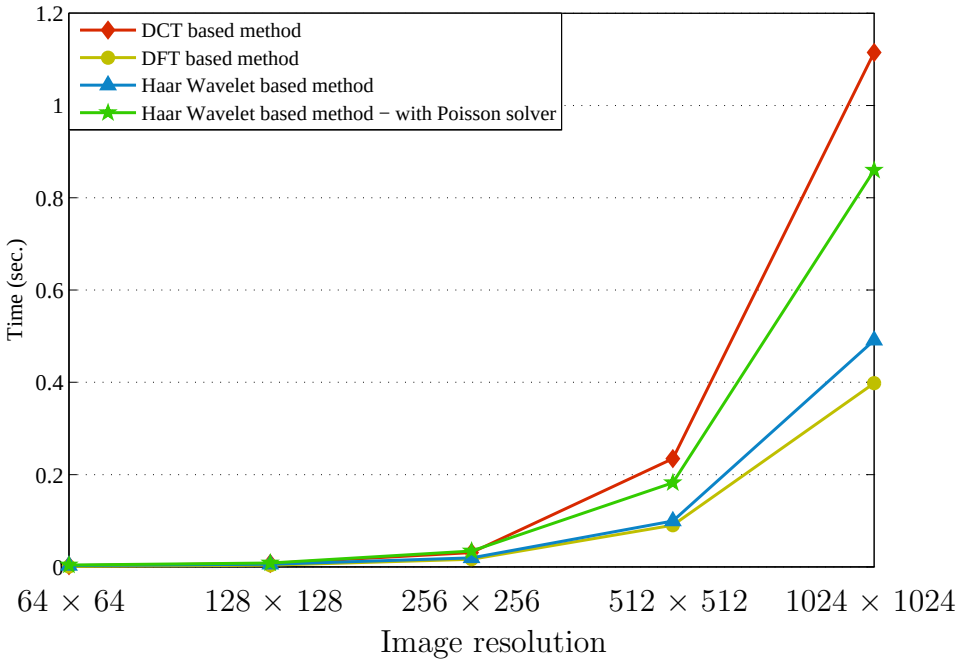


Figure 2.13: Reconstruction times versus image resolution

2.13 were obtained by running each algorithm 1000 times, measuring and recording the CPU time, and taking the average of all the values obtained per algorithm. This type of analysis is particularly employed when a single run of the code takes a fraction of milliseconds, as is the case herein. The averaging eliminates the fluctuation of computing time between individual runs and yields an increased degree of accuracy in the reported computation times. However, it should be stressed that these results should be interpreted considering also the core concepts of the studied algorithms, and how these concepts were implemented in the software used.

The core concept of the Frankot-Chellappa algorithm [16] is the discrete Fourier transform, as the method involves computations in the Fourier domain. The algorithm from [37] solves the Poisson equation in the DCT domain. The MATLAB implementation of the DCT transform relies on a fast Fourier transform built-in function. From this point of view then, the first two algorithms have similar computational complexity, and the evolution of computation time versus the number of pixels in the image should have a linearithmic behaviour: $\mathcal{O}(N \log N)$, where N is the number of samples of the considered signal. The integration method of Hampton et al. [24] is wavelet based. The computational complexity of wavelet based algorithms is linear with respect to signal size: $\mathcal{O}(N)$. A possible the reason why the data collected

does not seem to reflect this behaviour can be found in the technical details of the implementation.

The implementation of the Frankot-Chellappa method[2] employs MATLAB command `fft2`. The MATLAB command `fft2` is a built-in function that computes the 2D Fourier transform using a fast algorithm coded in C. The implementation of the DCT-based Poisson solver was done using the `dct` command, which is coded in MATLAB using the `fft2` function. This explains why this method performed worse than the Frankot-Chellappa algorithm in terms of CPU time. The entire implementation of the Hampton et al. method was done in MATLAB, and therefore it is unlikely that its speed outperforms an algorithm that uses only the `fft2` command, namely, the Frankot-Chellappa algorithm.

In summary, the simulations and the data collected show that the reconstruction method of Hampton et al. [24], with the elimination of the waffle pattern as described in this chapter, yields the best results in terms of solution accuracy in a comparable time with the fastest of the algorithms.

2.4.2 Reconstructing the image from noisy gradient data

In this section, the robustness to noise of the four algorithms is analyzed and compared. The performance of the reconstruction algorithms is evaluated on the same images considered in the previous section.

As noted, the test image `ramp peaks` and the other test images (shown in Figure 2.12) are significantly different. The image `ramp peaks` is representative for computer generated surfaces, has a low dynamic range and mainly low frequencies. As a result, its gradient is low in magnitude and both of its components have an overall smooth aspect, as illustrated in Figure 2.11. The other test images are representative for real digital images. Their dynamic range is the typical dynamic range of 8-bit encoded digital images and they are richer in high frequency content than the computer generated surface. The purpose of this discussion is to highlight the fact that the performance of the algorithms was evaluated in reconstructing a wide range of surfaces: rich in low frequencies, such as test image `ramp peaks` or rich in high frequencies, such as test image `barbara`.

The framework in which the analysis was performed is the following. As a starting point in this study, the situation when there is no noise in the input gradient was

[2]Downloaded from `http://www.umiacs.umd.edu/~aagrawal/software.html`

Table 2.3: Reconstruction from noisy gradient - test image `ramp peaks`

SNR in gradient	Relative error re			
	DCT Poisson solver [37]	Frankot-Chellappa [16]	Method from [24]	Method from [23]
12.4948	0.0446	0.1460	0.0515	0.0490
6.7726	0.1050	0.1533	0.1159	0.1058
3.1423	0.1370	0.1933	0.1550	0.1376
0.6593	0.1836	0.2068	0.2073	0.1926
-1.3178	0.1890	0.2169	0.2528	0.2124
-2.8539	0.2876	0.2359	0.3224	0.3021
-4.4620	0.3466	0.2944	0.3910	0.3652
-5.3237	0.3704	0.3163	0.4275	0.3982
-6.4405	0.4610	0.3960	0.5112	0.4780

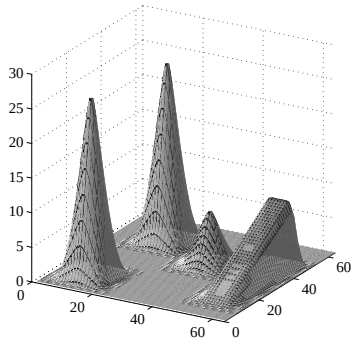
Table 2.4: Reconstruction from noisy gradient - test image `man`

SNR in gradient	Relative error re			
	DCT Poisson solver [37]	Frankot-Chellappa [16]	Method from [24]	Method from [23]
12.5173	0.0302	0.1999	0.0367	0.0333
6.4988	0.0597	0.2139	0.0735	0.0634
2.9743	0.0966	0.2087	0.1159	0.1008
0.4868	0.1189	0.2238	0.1512	0.1246
-1.4630	0.1422	0.2372	0.1787	0.1495
-3.0392	0.1935	0.2439	0.2243	0.1924
-4.3890	0.2236	0.2829	0.2663	0.2281
-5.5460	0.2275	0.2792	0.2865	0.2410
-6.5600	0.3003	0.3080	0.3563	0.3085

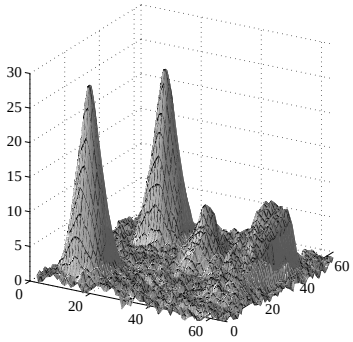
considered (resulting in an infinite SNR), and the results reported in Table 2.1 were confirmed. Then, the Hudgin aligned gradient of the original images was increasingly corrupted by Gaussian white noise to cover a wide range of signal to noise ratios. The lowest SNR value included in the analysis was of -7dB. Although this value corresponds to a severe degradation of the input gradient (as it is known, a negative value of SNR in dB indicates the fact that the signal of interest is less powerful than the noise), the purpose of including negative SNR values in the analysis was to evaluate the performance of the algorithms in these extreme conditions and verify if the reconstructions produced still represent reasonable estimates of the image.

The performance of the reconstruction algorithms is evaluated by analyzing the relative error of the reconstruction (formula 2.42) as a function of SNR in the input gradient. In Table 2.3 and 2.4 the results obtained for test image `ramp peaks` and `man` are displayed, respectively[3]. The reconstructions corresponding to the highlighted entries in the tables are shown in Figures 2.14 and 2.15, together with the original images. These reconstructions were chosen to visually illustrate the perfor-

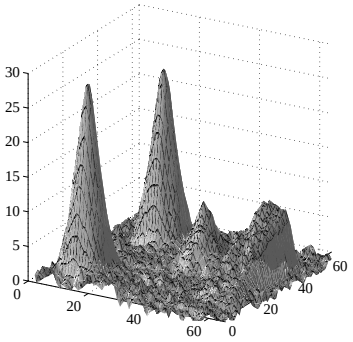
[3]Comprehensive tables containing all the results can be consulted at the end of this thesis, in Appendix A.



(a) Original image



(b) DCT based reconstruction [37]



(c) Haar wavelet based reconstruction [23]

Figure 2.14: Image reconstruction from noisy gradient: test image `ramp peaks`. SNR in input gradient: 0dB.

mance of the algorithms when signal and noise are equally dominant in the input gradient data set.

In Figure 2.17 a chart displaying the performance of the four tested algorithms on all test images is shown. As the graph indicates, for a low level of gradient degradation (indicated by a high value of the SNR), the performance of the DCT based method [37] and the performance of the two Haar wavelet based algorithms are similar. Overall, for values of SNR as low as -5dB, the DCT method performed the best. The reconstruction algorithm of Hampton et al. [23] with the Poisson solver included during the synthesis step offered comparable results, as can be seen from Figure 2.17 and from the data displayed in Table 2.5. The profile of the curve representing the Frankot-Chellappa algorithm seems to indicate that this algorithm produces better results for highly degraded input gradient, but unfortunately at this low level of SNR, even the best reconstruction does not offer an acceptable quality. An

(a) Original image

(b) DCT based reconstruction [37]

(c) Haar wavelet based reconstruction [23]

Figure 2.15: Image reconstruction from noisy gradient: test image `man`.
SNR in input gradient: 0dB.

example of this extreme case is illustrated in Figure 2.16. In summary, the simulations and the data collected indicate that for high values of SNR in the input gradient, the analyzed reconstruction algorithms produce similar results. For medium values of SNR in the input gradient, the best performance is obtained with the DCT-based Poisson solver [37], and the second best very close results are given by the method of Hampton et al. [23], with the Poisson solver included during synthesis. For extreme

(a) Original image

(b) DCT based reconstruction [37]

(c) Haar wavelet based reconstruction [23]

Figure 2.16: Image reconstruction from noisy gradient: test image `camera`. SNR in input gradient: -6.5dB.

situations of very low values of SNR in the input gradient, the best performance is given by the DFT based Frankot-Chellappa algorithm, followed by the DCT based Poisson solver and then by the Hampton et al. method [23].

2.5 Conclusions

The problem of reconstructing an image from a given gradient data set was presented. The Haar wavelet based reconstruction method of Hampton et al. [23,24] was described in detail. An extension to the original algorithm was described, to improve

Table 2.5: Image reconstruction from noisy gradient - centralizing chart obtained by averaging the relative errors of the reconstructions obtained from similar values of SNR in input gradient for all tested images

SNR in gradient	Relative error re			
	DCT Poisson solver [37]	Frankot-Chellappa [16]	Method from [24]	Method from [23]
12.5506	0.0348	0.1596	0.0413	0.0382
6.6208	0.0754	0.1686	0.0871	0.0784
3.0643	0.1084	0.1824	0.1263	0.1127
0.5663	0.1372	0.1931	0.1643	0.1443
-1.3720	0.1693	0.2115	0.2081	0.1797
-2.9298	0.2243	0.2345	0.2563	0.2300
-4.3719	0.2657	0.2662	0.3074	0.2767
-5.4551	0.2984	0.2758	0.3523	0.3130
-6.4903	0.3515	0.3161	0.4019	0.3630

the reconstruction results, given the Hudgin alignment of the gradient. Two types of the Haar wavelet based reconstruction method were compared with two other reconstruction techniques. In the analysis, the performance of the four algorithms was evaluated by comparing the accuracy of the result produced and the computational speed.

To evaluate the accuracy of the results obtained, the difference between the reconstructions obtained and the original image was studied, by comparing the relative error under two distinct scenarios. First, the ideal case of obtaining an estimate of the given image from a perfectly modelled, noise-free gradient was considered. Then, the more realistic case of reconstructing the image from noise corrupted gradient data set was studied.

In the case of almost no noise in the input gradient, the best results were produced by the technique of Hampton et al., with the modification proposed in section 2.3.2. In the case of medium degradation of the input gradient, the best performance was obtained with the DCT based Poisson solver [37], and the second best very close results were given by the method of Hampton et al. [23], with the Poisson solver included during synthesis. In the case of severe degradation of the input gradient, the best performance was obtained using the DFT based Frankot-Chellappa algorithm, followed by the DCT based Poisson solver and by the Hampton et al. method [23].

The four reconstruction algorithms have different computational complexities. The DCT and DFT based methods are $\mathcal{O}(N \log N)$, whereas the Haar wavelet based method is $\mathcal{O}(N)$. Measurements of CPU time necessary to obtain an estimate of the image from a given gradient set indicated that for lower sizes of the reconstructions (128×128 to 256×256), the performance of the four algorithms was similar. For

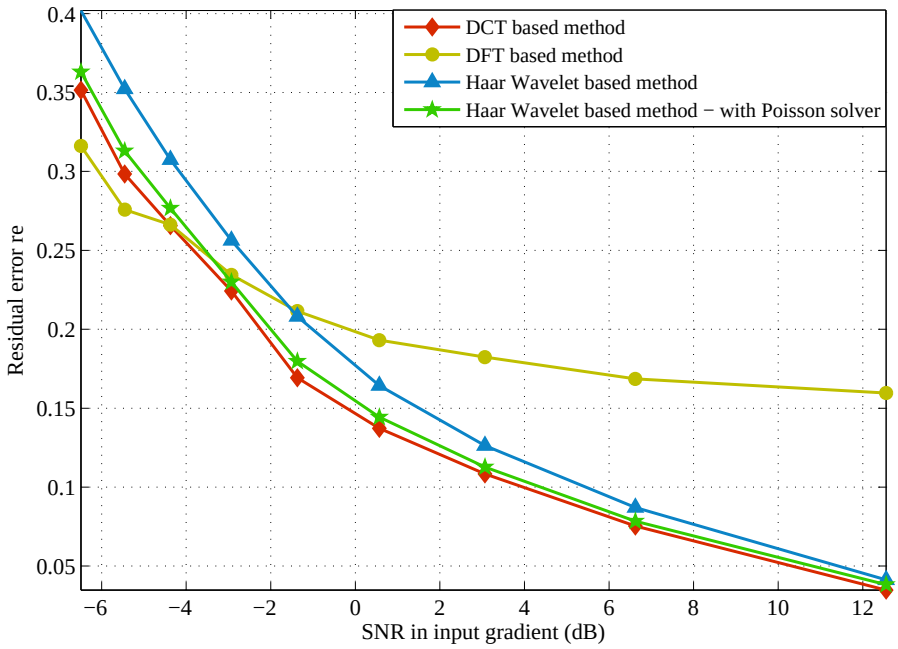


Figure 2.17: Relative reconstruction error versus SNR in input gradient

images of higher dimensions (256×256 to 1024×1024), in the current implementation, the Frankot-Chellapa reconstruction technique [16] produced the results in the shortest time. The reconstruction method of Hampton et al. [24] with the modification proposed in this chapter ranked the second. Adding a Poisson solver during the synthesis stage slowed the algorithm, but still the results were obtained faster than using the DCT based Poisson solver [37].

Chapter 3

Applications of image processing in the gradient domain

3.1 Introduction

In this chapter, a stitching and object insertion techniques are presented as possible approaches to these applications. In its most general case, the stitching method has three main stages: an image registration stage, a blending stage, and an image reconstruction from gradient stage. In the registration stage, the images to be stitched are spatially aligned, using a Mexican hat wavelet based technique [49]. During the blending stage, the gradients corresponding to the common region of the input images are combined. The object insertion method presented in this chapter assumes the images to be combined already registered and relies on combining the gradients of a region of interest in the two images to generate a new gradient from which the composite image is recovered. In both techniques, the final image is reconstructed from the gradient using the method of Hampton et al. [23,24], with the Poisson solver included during the synthesis stage.

The organization of this chapter is as follows. In section 3.2, the problem of image stitching is introduced and the main facets of the problem are described. In section 3.3 a stitching method is developed. Next, in section 3.4, a seamless object insertion method is presented, in the context of image morphing. In both techniques, the final image is reconstructed from a gradient data set using the method proposed by Hampton et al. [23, 24].

3.2 Image stitching

Large view field images are of great importance in many practical applications. In areas ranging from magnetic resonance imaging to satellite imagery, a computationally efficient and easy to implement method to produce high resolution wide angle images is of utmost importance and developing such algorithms continues to draw research interest. Aside from the physical solution of introducing wider lens in the imaging device, which may be difficult to implement on a large scale and also more expensive, image stitching algorithms represent the typical alternative.

In image stitching (also known as *image mosaicking*), multiple parts of a scene are combined into a single large representation of the same scene. Perhaps the most common application of image stitching is producing panoramic images (also referred to as *mosaics*) from partially overlapping digital representations of a scene. These representations are generally digital photographies taken at the same time from different locations, or at different times, with the same or with different cameras, situated at the same or different distances from the scene. All these different conditions generate distortions in the representations of the scene, which can be light or color inconsistencies or geometrical deformations. Existing image stitching techniques generally operate on image intensity, in the gradient domain, or at various resolution levels.

Depending on the scene content, on the relative position of the camera with respect to the imaged scene and on the intended application, several main stages are identified and addressed by stitching algorithms. First, the images to be combined are spatially aligned. This stage is called *image registration*. Next, a larger image containing all the registered images is defined and the light or color differences visible at the boundaries between different input images are removed, resulting in a seamless mosaic image.

This stage is called *image compositing and blending*. In Figure 3.1 the main stages of image stitching are illustrated by a simple example of obtaining a mosaic from two partially overlapping input images [1].

3.2.1 Image registration

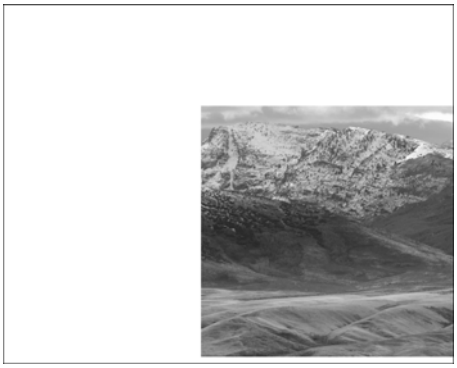
Image registration has applications in object recognition, panoramic, satellite and biomedical imaging and is also a key stage in image stitching algorithms. During image registration, the images from which the mosaic will be produced are spatially

[1] A color version of the input images is available at http://www.pbase.com/ckuhn55/wyoming/

(a) Image 1

(b) Image 2

(c) Intermediate image 1



(d) Intermediate image 2

(e) Seamless mosaic

Figure 3.1: Obtaining a seamless mosaic

aligned. In many practical applications, the digital images used to obtain a mosaic of a certain scene are taken with different cameras from different viewpoints. These differences in the initial setting produce differences in scale and orientation in the set of images from which the mosaic must be obtained. Typically, during the registration step, one of the input images is denoted as the *reference image*, and the objective is to find a geometric transformation which maps the other images onto the coordinate system of the reference image.

Various feature based registration algorithms have been reviewed in [21]. Feature based registration techniques generally begin with a selection of features in the images to be registered. The type of features varies from application to application and depends on the information contained in the considered image. Typical examples of image features include points, corners, curves, regions, and patterns [8].

A wavelet based image registration method is proposed in [49], in the context of aerial images registration. The method is further shown to have applications in face recognition in [20]. In this chapter, this registration algorithm is applied as a first stage in the proposed image stitching method.

In what follows, an overview of this registration algorithm is presented. The registration technique considers the case of spatially aligning two images and consists of three main steps. First, a set of feature points is identified in the given images. Then, a correspondence between these feature points is established. Finally, the coordinates of the matching feature points are used to find the registration transform, using an iterative weighted least squares algorithm.

The feature points are identified independently in the two images considered. The identification of a feature point of an image is achieved as follows: the image is filtered with two scaled versions of a Mexican-hat wavelet, and the location of local maximum in the difference between the two responses is declared a feature point. The feature point identification step is repeated for a range of scales for each image, to address the case when the input images are taken from different distances from the imaged scene (i.e., the zooming factor in the input images is different). The bounds of the range of scales are two user specified parameters, related to the zooming difference between the images.

As a second stage in the algorithm, a correspondence between the identified feature points is established. The correspondence is found by computing and comparing Zernike moments of circular neighbourhoods around the previously identified feature points. The radius of the circular neighbourhood in which the search is conducted

(a) Visible seam generated by photometric inconsistency between the input images

(b) Discontinuous contours generated by previous registration errors

Figure 3.2: Typical artifacts in image stitching

is automatically computed from the range of scales previously established. Once the Zernike moments are computed, a pair of feature points is declared a match if their corresponding Zernike moments are similar in magnitude (i.e., if the difference in magnitude is below a threshold). The Zernike polynomials provide an appropriate tool for the task at hand, in particular due to the rotational symmetry property of their magnitude, desirable in the context of image registration.

As a final step in the registration technique of [50], an iterative weighted least-squares minimization is performed to exclude the pairs of points falsely identified as a match and to determine a set of parameters from which the desired affine registration transform is obtained.

The registration technique is shown [49, 50] to produce good results even in the case of geometric distortions such as differences in scale between the two input images, or additional degradations such as noise or blurring. Therefore, this technique was chosen to perform the registration in the proposed image stitching algorithm.

3.2.2 Image compositing and blending

The images from which the mosaic will be obtained are spatially aligned during the image registration stage. Upon completion of this stage, an initial mosaic consisting of the "union" of all component images can be defined. If a simple "cut and paste" operation is employed directly on the intensities of the input images, this initial mosaic usually contains visual artifacts caused either by different lighting conditions (also known as *photometric inconsistencies*) or registration errors. Different lighting

conditions generate seams at the edge between two input images, as can be seen from Figure 3.2a, whereas registration errors generate unnatural "jumps" in the contours of the objects in the mosaic, as illustrated by Figure 3.2b.

The presence of these visual artifacts in the initial mosaic indicates the necessity of a next stage in stitching algorithms, a stage in which further processing yields a natural looking mosaic image, without any seams or discontinuous contours. This stage of image stitching techniques is called "image compositing and blending" [40] and handles precisely the above mentioned type of artifacts.

In this stage, first the surface on which the images will be combined has to be defined, by specifying its shape and size. Then, a set of intermediate images is obtained, by representing the reference and registered images in the newly defined surface. All of the intermediate images have the same size and shape as the desired mosaic. Finally, the mosaic image is produced by combining these intermediate images into a visually pleasing and natural looking image.

In image blending, the objective is to determine a function to combine the intermediate images in such a way that in the final mosaic the transition from one component image to the other is imperceptible. Good blending techniques should produce seamless mosaics by compensating for existing exposure or color differences between the component images and should introduce as little distortion as possible in the overall result. Previously proposed image blending algorithms operate on image intensities or on image gradient, at full resolution of the image or at multiple resolution scales.

A well known intensity based blending technique is feathering [39], where the mosaic is generated by computing a weighted average of all intermediate images. In the composite mosaic image, pixels are assigned weights proportional to their distance from the centre of the image they come from, resulting in a smoother transition from one image to the other in the final mosaic.

One of the first intensity based multiscale blending methods is presented in [10] and relies on pyramid decomposition. The intermediate images are decomposed into bandpass filtered components and the blending is done at each scale, in a transition zone inversely proportional to the spatial frequency content in the band. In a similar approach [26], pyramid decomposition is replaced with wavelet decomposition.

In this work, a blending function is defined to combine the gradients of the intermediate images and produce a gradient data set from which the mosaic image needs to be reconstructed. Generally, gradient modifications produce a data set which is not

a conservative vector field and image reconstruction from this data set no longer has an exact solution. The problem of image image reconstruction from a gradient data set was largely described in chapter 2. One of the reconstruction techniques presented therein will be employed as a final stage in the stitching algorithm presented in the following section.

3.3 Seamless image stitching in the gradient domain

The stitching method presented here yields seamless image mosaics and is based on combining the gradients of two input images to generate a set of gradients for the mosaic image. Then, the mosaic image is obtained from this gradient using the reconstruction technique of Hampton et al. [23, 24], described in the previous chapter. The stitching method presented here was proposed in [36]. In this section, this stitching method is described and further developed to address several situations of gradually increasing difficulty. First, the case in which the images to be stitched are in grayscale representation and have a rectangular overlap region is considered. Then, the situation in which the images to be stitched are in grayscale representation and have an irregularly shaped overlap region, with an unknown location, is examined. Finally, the case when the images to be stitched are in color representation and have an irregularly shaped overlap region, is taken into consideration and a method to stitch them into a seamless mosaic is proposed.

3.3.1 Stitching of two grayscale images with rectangular overlap region

Consider two registered input images Φ_1 and Φ_2 with a common overlapping region, as illustrated diagrammatically in Figure 3.3. The objective is to combine the two images into a seamless mosaic. The gradients $\tilde{\Phi}_1^F$ and $\tilde{\Phi}_2^F$ of the input images are computed and stitched together to generate a gradient data set from which the mosaic image will be reconstructed. In the overlapping region, the gradients are blended using a weighting function, while in the rest the gradients are simply stitched together.

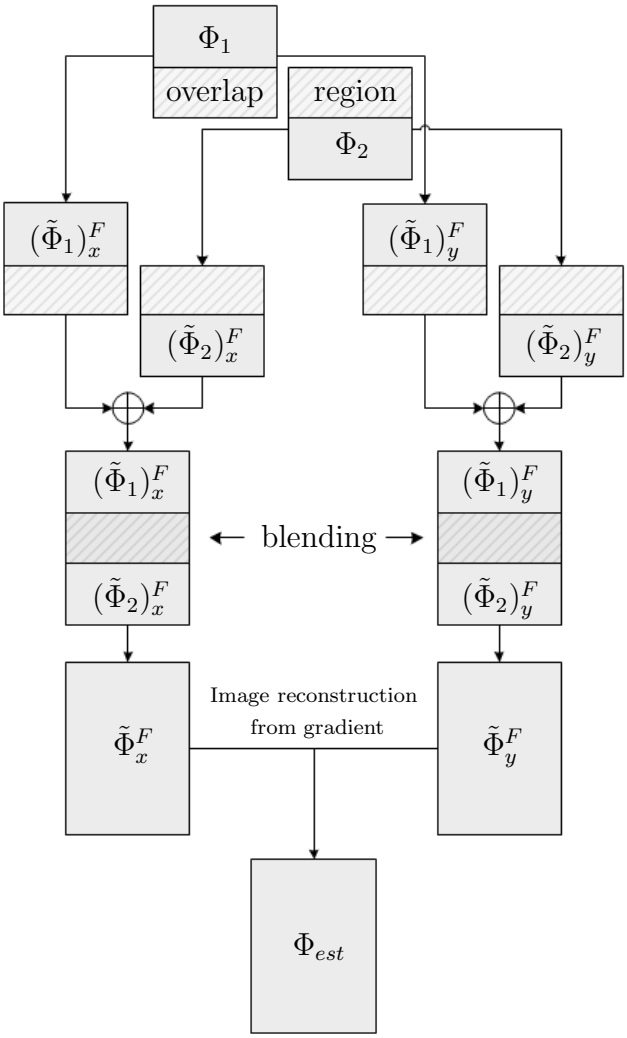
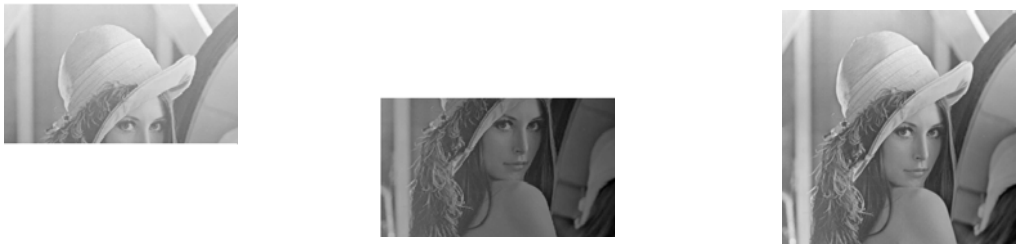


Figure 3.3: Overview of seamless stitching method - rectangular overlap region



(a) Input image Φ_1 **(b)** Input image Φ_2 **(c)** Mosaic obtained with proposed algorithm

Figure 3.4: Stitching two images: Lena example

The blending of gradients in the overlap region is described by the following equation:

$$\tilde{\omega} = w(x, y) \cdot \tilde{\omega}_1 + [1 - w(x, y)] \cdot \tilde{\omega}_2 \tag{3.1}$$

where $\tilde{\omega}_1$ and $\tilde{\omega}_2$ denote the gradients of Φ_1 and Φ_2 in the overlap region, respectively, and $w : \mathbb{R}^2 \to [0, 1]$ is a linearly or exponentially decreasing function.

This simple approach results in the gradient from the upper image dominating the upper part of the overlapping region while the lower gradient dominates the lower part. The gradient of the mosaic image outside the overlapping region is defined to be equal to the gradient from the non-overlapping parts of the input images, respectively. The mosaic image is obtained from the stitched gradient using the Hampton et al. reconstruction method described in section 2.3.2, with the Poisson solver included during the synthesis stage. The mean value of the image to be reconstructed is estimated as the average intensity of the input images.

The proposed method can thus be summarized as follows:

1. Compute the gradient of the input images Φ_1 and Φ_2
2. Create the stitched gradient using the blending function given by 3.1 in the overlap region and "cut and paste" in the rest to obtain $\tilde{\Phi}$
3. Obtain a mosaic image from $\tilde{\Phi}$, using the Haar wavelet reconstruction technique of Hampton et al. [23] described in chapter 2, with the Poisson solver at each resolution level.

In Figure 3.4 a mosaic image is obtained from the two images shown in Figures 3.4a and 3.4b, using the new stitching method. The performance of this method will be discussed with examples in section 4.1.1 of the next chapter.

3.3.2 Stitching of grayscale images with irregularly shaped overlap region

In the previous section, a method for seamless stitching of images was presented. The method assumed the input images to be in an arrangement as in Figure 3.3. In practice, however, the arrangement of the input images is often more challenging than this. For instance, a more frequently encountered arrangement of the input images is illustrated in Figure 3.5. The region in which the two input images overlap

Figure 3.5: Irregularly shaped overlap region.

(a) Intermediate image Ψ_1.

(b) Intermediate image Ψ_2.

Figure 3.6: Stitching aerial images. The white regions surrounding the images are outside the region of support.

has an irregular shape and the weighting function used to combine the gradients in the overlap region cannot be defined as straightforward as in section 3.3.1. For comparable results (visually), in this case, the weighting function is replaced with an averaging operation.

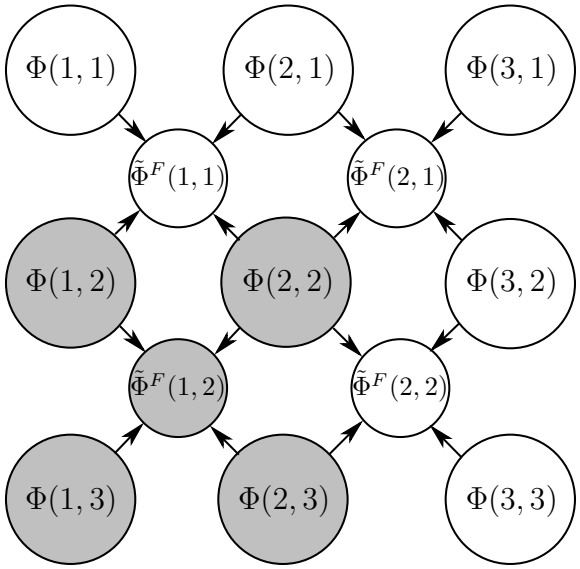


Figure 3.7: Error in gradient cause by padding in the intensity domain. Large circles: intensity values. Small circles: gradient values. Gray: meaningful information, from the original image. White: erroneous information, from padding. The arrows indicate which pixels contribute to a certain gradient in Fried alignment.

The modified stitching method is presented in what follows. Given two images to be combined and a registration transform, first, two intermediate images of the same size as the final mosaic are produced. The two images are obtained from the reference image and the registered image, by padding with a constant value to the desired shape and size. In Figure 3.6 this stage of the stitching method is illustrated.

In the following stage of the algorithm, the Fried directional derivatives of the intermediate images Ψ_1 and Ψ_2 are computed. Because of the way in which the intermediate images were obtained (namely, by padding with a constant value until the desired size is reached), an error is introduced in all gradient values computed from pixels situated on the border of the region of support, regardless of the geometry considered. To understand the cause of this error, consider the diagram in Figure 3.7. In Figure 3.7, the large circles represent pixel (intensity) values. The small circles represent Fried gradient values. A large gray shaded circle means that the pixel value at that point contains useful image information from one of the original images to be stitched. A large white circle means that the pixel value at that point is equal to the constant value used in the padding stage. As was discussed in section 2.2.1 of chapter 2, in the Fried discretization model, four neighbouring pixels of the image contribute to one value of the directional derivative, as indicated by the orientation

of the arrows in Figure 3.7. Therefore, of the four gradient values (small circles) shown in Figure 3.7, only the bottom left is meaningful and should be used during the reconstruction. In light of this example, we can conclude that all along the border of region of support of the intermediate images, the gradient values are unreliable and should not be used in the reconstruction. The approach herein is to first identify the location of the unreliable gradients, and then set their value to zero in the gradient data set from which the composite image will be reconstructed.

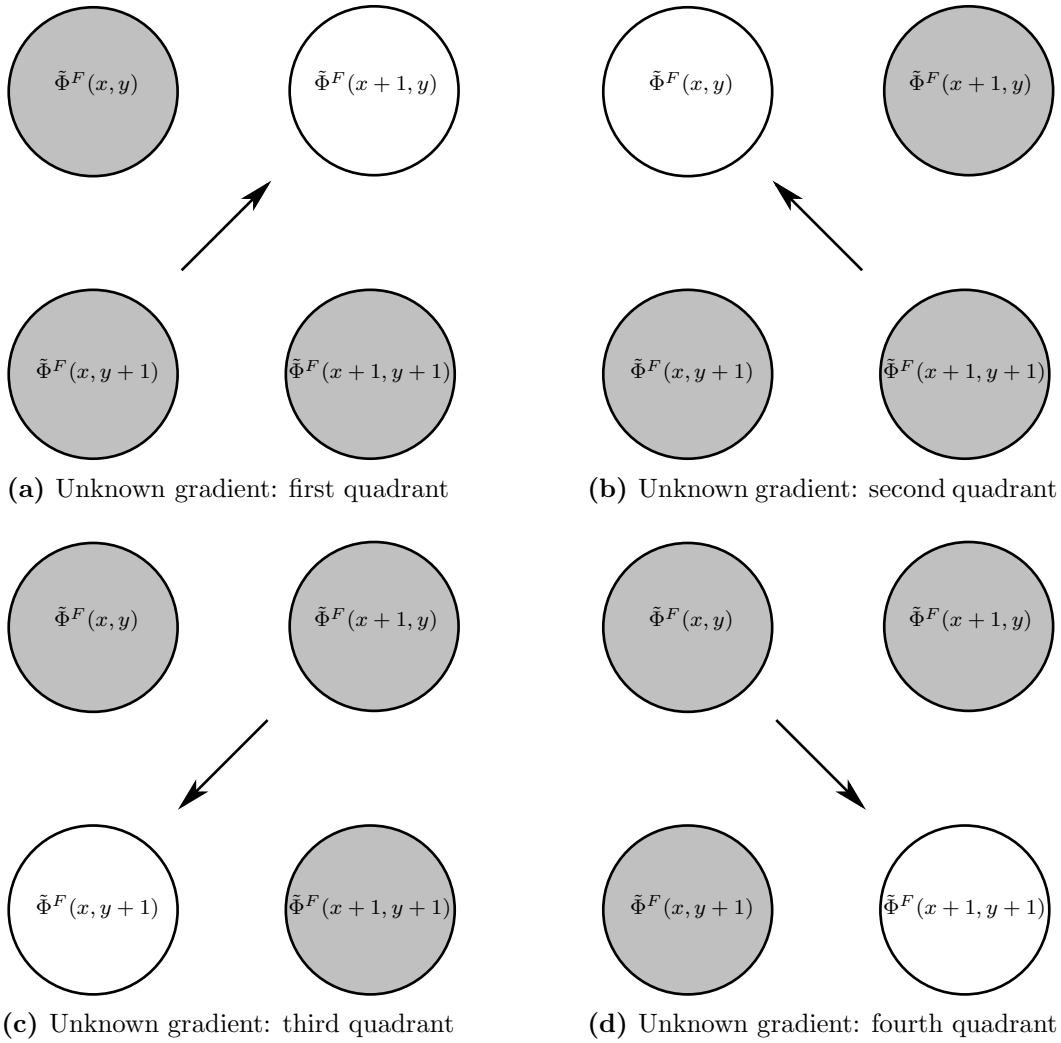
The gradient data set from which the mosaic will be obtained is defined in two steps. An initial gradient data set is described by:

$$\tilde{\Phi}^F(x,y) = \begin{cases} \tilde{\Psi}_1^F(x,y) & \text{, for } (x,y) \in \Omega_1 \backslash \Omega_2 \\ \tilde{\Psi}_2^F(x,y) & \text{, for } (x,y) \in \Omega_2 \backslash \Omega_1 \\ 0.5\left(\tilde{\Psi}_1^F(x,y) + \tilde{\Psi}_2^F(x,y)\right) & \text{, for } (x,y) \in \Omega_1 \cap \Omega_2 \\ 0 & \text{, else} \end{cases} \tag{3.2}$$

where Ω_1 and Ω_2 denote the regions of support of the reliable gradient of the intermediate images shown in Figures 3.6a and 3.6b, respectively. The symbols $\backslash$ and $\cap$ denote set difference and intersection, respectively, and are defined here for reference.

$$A \backslash B = \{x : x \in A \text{ and } x \notin B\}.$$
$$A \cap B = \{x : x \in A \text{ and } x \in B\}.$$

However, the gradient defined by Equation 3.2 is still not an integrable data set. In order for a vector field to be the gradient of a scalar field, the zero curl condition must be satisfied (recall section 2.2.2, equation 2.14). In 2D, this corresponds to the symmetry of the second order partial derivatives. Even with the elimination of the unreliable gradient values, the zero curl condition is not satisfied at the boundary of the region of support of the data defined by Equation 3.2. One way of overcoming this problem and ensuring integrability of the data set along the boundary of the region of support is by extrapolating the reliable values of the gradient data outside its region of support, via *diagonally flat extrapolation* [25, 32]. The diagonally flat gradient extrapolation procedure was introduced in the context of wavefront reconstruction from incomplete, circular aperture gradient measurements and it is described below for reference.



(a) Unknown gradient: first quadrant

(b) Unknown gradient: second quadrant

(c) Unknown gradient: third quadrant

(d) Unknown gradient: fourth quadrant

Figure 3.8: Diagonally flat extrapolation of gradient. Gray circles represent given Fried gradient data. White circles represent unknown Fried gradient data.

On a four point gradient grid with three given values and one unknown value, one way in which integrability is ensured is by defining the fourth gradient value in such a way that the zero curl condition is satisfied. In Fried geometry, the curl condition for $\tilde{\Phi}^F$ can be written as follows:

$$\begin{aligned}&\tilde{\Phi}^F_x(x,y+1) - \tilde{\Phi}^F_x(x,y) + \tilde{\Phi}^F_x(x+1,y+1) - \tilde{\Phi}^F_x(x+1,y) = \\ &= \tilde{\Phi}^F_y(x+1,y) - \tilde{\Phi}^F_y(x,y) + \tilde{\Phi}^F_y(x+1,y+1) - \tilde{\Phi}^F_y(x,y+1)\end{aligned} \tag{3.3}$$

In Figure 3.8, four possible situations are illustrated. The location of the gradient

to be extrapolated is marked graphically by a white circle, and the locations of the correct (reliable) gradients are marked graphically by gray shaded circles.
The value of the fourth unknown gradient is found by solving equation 3.3. Regardless of the location of the gradient to be extrapolated, equation 3.3 has two unknowns (namely, the two directional derivatives at that point), therefore it is underdetermined and cannot be solved directly.

A possible approach to solve the equation 3.3 is by introducing an additional constraint on the two components of the unknown gradient value. In diagonally flat extrapolation, this constraint assumes the following forms, depending on the location of the gradient to be extrapolated, as defined in Figure 3.8:

$$\begin{aligned}
\tilde{\Phi}_y^F(x+1,y) &= \tilde{\Phi}_x^F(x+1,y), \text{ for quadrant I} \\
\tilde{\Phi}_y^F(x,y) &= -\tilde{\Phi}_x^F(x,y), \text{ for quadrant II} \\
\tilde{\Phi}_y^F(x,y+1) &= \tilde{\Phi}_x^F(x,y+1), \text{ for quadrant III} \\
\tilde{\Phi}_y^F(x+1,y+1) &= -\tilde{\Phi}_x^F(x+1,y+1), \text{ for quadrant IV}
\end{aligned}$$

By introducing these constraints, equation 3.3 can be solved in any of the situations illustrated graphically in Figure 3.8. The extrapolation process is then carried out successively, until the smallest rectangular area that encloses the region of support of the gradient is filled in.

Once the gradient set of the mosaic has been defined, the mosaic image is obtained using the Haar wavelet based reconstruction technique described in section 2.3.2. A Poisson solver is included at each resolution level during the synthesis stage, for increased smoothness of the resulting mosaic image.

In the general case, the images to be stitched are rectangular. In the proposed framework, stitching two rectangular images will yield a rectangular shape image, which encloses tightly the region of support of the mosaic. As specified at the end of section 2.3.2, when the size of the image of interest is not square ($2^M \times 2^M$), the image must be corrected after it is cropped from the initial reconstruction, by an amplitude shift which produces a desired mean value. The approach in the method proposed herein is to choose the desired mean value of the reconstruction equal to the average of the mean intensities on the two images to be combined. For example, two input images with mean values m_1 and m_2 will produce a mosaic image with mean value $0.5(m_1 + m_2)$.

(a) Input image 1

(b) Input image 2

Figure 3.9: Stitching images with irregularly shaped overlap: aerial example - input images.

(a) Intensity stitching

(b) Gradient stitching

Figure 3.10: Stitching images with irregularly shaped overlap: aerial example - results.

To illustrate the stitching method introduced in this section, in Figure 3.10b a mosaic image obtained from the two input images in Figures 3.9 is shown. The performance of the method will be discussed in section 4.1.2 of the next chapter.

3.3.3 Stitching of color images

In the more general and more frequently encountered in practice situation, the images of interest are given in color representation. A standard model to represent color images is the RGB model. In the RGB model, a pixel is uniquely determined by five parameters: two spatial coordinates and three values of the Red, Green and Blue

channel information, respectively, which combined define the color of the pixel. In current standards, an average digital photography has a resolution of 2 megapixels (MP), which means the average image has 2 million pixels. Given the complexity of registration and blending algorithms, a natural question is what is the most efficient way in which the problem of color image stitching should be addressed.

Each channel in the color representation of an image can be viewed as an approximation of the grayscale version of the image. The registration technique employed in the proposed stitching method is feature point based and the most computationally intensive part in this stage of the algorithm is finding the parameters that define the registration transform. With this in mind, the approach to handle the registration stage of combining the two images is the following. First, the parameters defining the registration transform are identified for the luminance channels of the images to be combined. Then, the registration transform is obtained from these parameters and applied on each channel of the color representation of the images.

The blending method as described in this chapter is gradient based. As was discussed in the end of section 2.2.1 in chapter 2, the gradient discretization models presented therein extend naturally in the case of color images. Therefore, it seems reasonable to propose an extension of the blending method in the case of combining multi-channel images. Unfortunately, unlike in the registration case, blending only luminance information does not produce visually plausible results and neither does blending only one channel from the RGB representation of the two images. This is to be expected, given the continuity of a meaningful image as a function of color, and the nature of the color representation employed.

A solution for blending the previously registered images into a seamless visually plausible color mosaic is presented in what follows. As a preprocessing step, the input images are split into their three component channels. Then, the gradients of the corresponding channels are computed and stitched together as described in section 3.3.2. At the end, the three channels of the desired mosaic are reconstructed from the stitched gradients.

In the most general case the images to be stitched are rectangular. As specified at

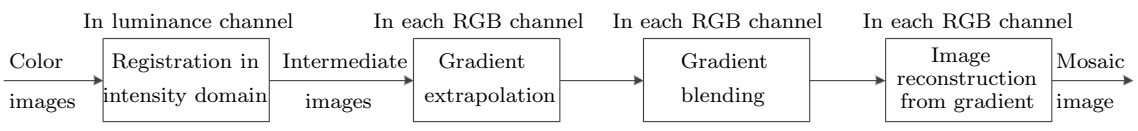
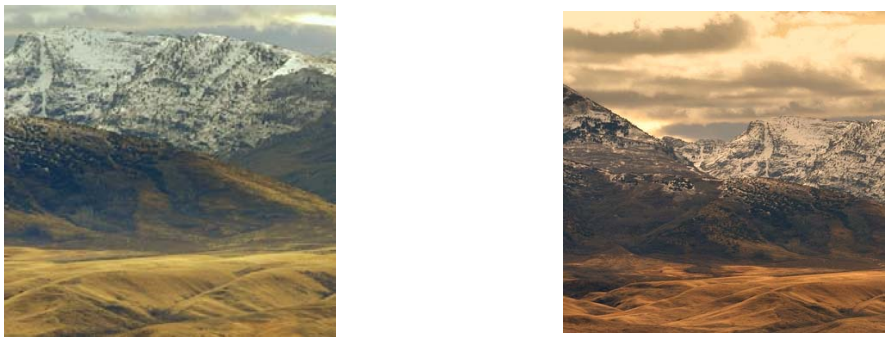


Figure 3.11: Overview of stitching algorithm for color images



(a) Input image 1 **(b)** Input image 2

Figure 3.12: Input images

(a) Intensity stitching

(b) Gradient stitching

Figure 3.13: Stitching color images

the end of section 2.3.2, in this case the reconstructed image signal must be corrected by an amplitude shift which produces a desired mean value. The approach in the method proposed herein is, for each channel, to choose the desired mean value of the reconstruction equal to the average of the mean intensities on the two images to be combined, in the respective channel. For example, two input images with mean values (r_1, g_1, b_1) and (r_2, g_2, b_2) will produce a mosaic image with mean values $0.5(r_1+r_2, g_1+g_2, b_1+b_2)$. An overview of the algorithm is shown in Figure 3.11 and in Figure 3.13 a mosaic image is obtained from the two images shown in Figure 3.12. For comparison purposes, in Figure 3.13a, the result obtained by stitching intensities is shown. Here, the stitching algorithm operates directly on each RGB channel, as opposed to combining their gradients, and this produces visible seams and noticeable color differences in the composite image.

A more detailed discussion concerning the performance of the method will be illustrated with examples in section 4.1.3 of the next chapter.

3.4 Seamless object insertion

An interesting image processing problem is image morphing. Image morphing denotes the technique used in special effects in movies or animation to gradually modify an image (or a part thereof) into another, through a seamless transition. In movies or animated films, morphing an image into another is obtained by displaying a sequence of frames in which the initial image gradually changes into the desired one [35, 48]. For a plausible effect, in each of the individual frames the transition from one image to the other must be natural and without any seams or other visible artifacts.

A possible approach to create such images is known as *object insertion*. The object insertion problem admits the following straightforward formulation. Given two images, the goal is to partially replace the content of one image with a desired content from the other image, in a seamless way.

Previous research [11, 31] indicates that the gradient domain provides a suitable framework for the application at hand. In what follows, a conceptually simple object insertion method operating in the gradient domain is used. Specifically, the gradient of a composite image is obtained inserting a desired content from the gradient of an image into a region of interest from the gradient of another image. This approach is also used by Perez et al. [31]. What is different here is that the image is then reconstructed from this gradient using the Haar wavelet based reconstruction tech-

nique earlier reviewed, with a Poisson solver included at each resolution during the synthesis step.

The method presented in this section is applicable in the general case of color images. Unless otherwise specified, here the gradient is to be regarded as a two dimensional vector in which each component is a 3D array, representing the directional Fried derivative of each of the Red, Green and Blue channels of the respective image. The image from which the content is copied to will be referred to as the *source* image and the image into which the content is copied will be referred to as the *target* image. The location where the content will be copied from the source image to the target image will be referred to as *region of interest*.

To illustrate the problem of object insertion and the terminology, consider Figure 3.14. The source and target images are Φ_1 and Φ_2, and can be seen in Figure 3.14a and 3.14b, respectively. The goal is to replace the current content of the region R_2 from the target image Φ_2 with the content of the region R_1 of the source image Φ_1.

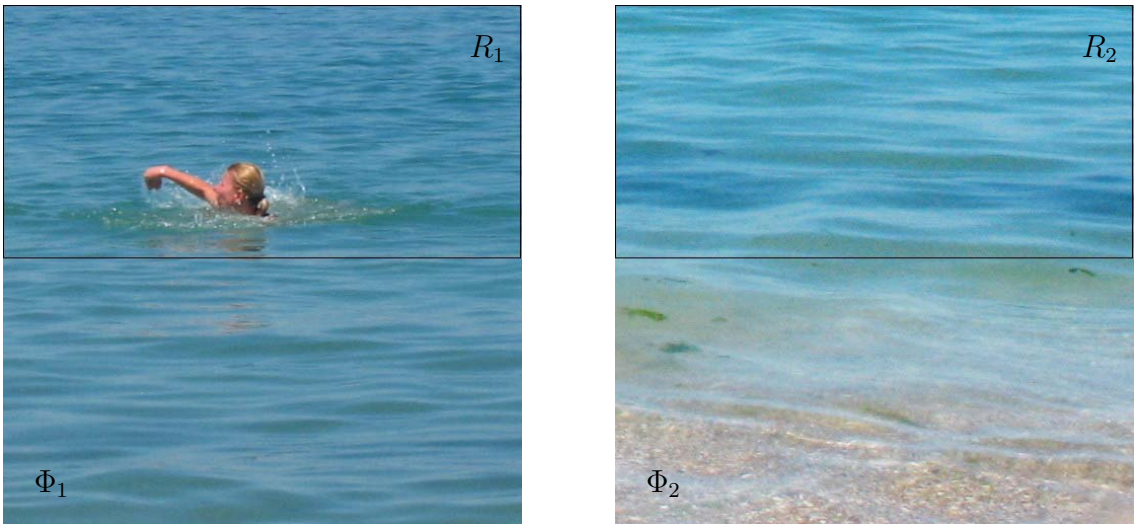
The approach presented herein is the following. First, the Fried gradients $\tilde{\Phi}_1^F$ and $\tilde{\Phi}_2^F$ of the two images are computed. Then, the gradient of the composite image is defined by:

$$\tilde{\Phi}^F(x,y) \quad = \begin{cases} \tilde{\Phi}_1^F(x,y) & \text{, for } (x,y) \in R_2 \\ \tilde{\Phi}_2^F(x,y) & \text{, for } (x,y) \in \Phi_2 \backslash R_2 \end{cases} \tag{3.4}$$

That is, the gradient data set of the composite image is created by replacing the gradient data corresponding to the region of interest R_2 in the target image with the gradient data corresponding to the region R_1 from the source image. The composite image is obtained from this gradient data set using the Haar wavelet based reconstruction technique described in section 2.3.2, with the Poisson solver included at each resolution level during the synthesis step. As discussed earlier, the processing is done in each channel of the color representation of the images. A result obtained with the presented method is shown in Figure 3.15 and the performance of the method will be discussed in section 4.2 of the next chapter.

3.5 Conclusions

The problem of image stitching was presented, and several existing approaches to obtain seamless mosaic images were reviewed. A new stitching algorithm was intro-



(a) Source image **(b)** Target image

Figure 3.14: The problem of object insertion

Figure 3.15: Seamless object insertion: result.

duced. In its most general case, the method has three main stages: image registration, gradient based blending and wavelet based image reconstruction from the gradient. The proposed algorithm is described in three different scenarios of gradually increasing difficulty.

A conceptually simple object insertion method was presented. The method is gradient based and relies on replacing the gradient information from a region of

interest of a target image with the gradient information from a region of interest of a source image.

In both analyzed applications, the final composite image is obtained from a gradient data set using the Haar wavelet based reconstruction technique of Hampton et al. [23, 24], with the Poisson solver included at each resolution level during the synthesis step.

In the following chapter, the performance of the new image stitching and object insertion techniques is illustrated with examples under different conditions.

Chapter 4

Experimental results

The objective in image stitching or morphing applications is to produce visually plausible composite images, from a set of partially overlapping input images. In this chapter, operations in the gradient domain are compared with operations in the intensity domain and the results prove that the gradient domain provides a more adequate framework for the analyzed applications.

The organization of this chapter is as follows. In section 4.1, the stitching method proposed in section 3.3 is applied to produce seamless mosaics. The performance of the technique is evaluated by analyzing the results produced using input images representative for several scenarios of increasing level of difficulty. A gradient domain based object insertion technique is analyzed in section 4.2 in the context of image morphing and several examples are presented. In section 4.3 a discussion regarding the criteria of performance evaluation is presented and in section 4.4 the content of the chapter is reviewed and the main conclusions are drawn.

4.1 Image stitching

In this section, the performance of the stitching method proposed in chapter 3 is evaluated to produce mosaic images from various input images. The testing set of input images is described and the results obtained are analyzed and compared. The section is organized as follows. First, the method is tested on stitching to grayscale images with a rectangular overlap region. Then, the performance of the algorithm is studied in the situation when the images to be stitched are in grayscale representation and have an arbitrary shaped overlap region. Finally, in a more general case, the

(a) Photometric inconsistency artifact

(b) Misregistration artifact

Figure 4.1: Typical artifacts in image stitching

method is applied to combine two color images into a seamless mosaic.

4.1.1 Stitching of two grayscale images with rectangular overlap region

Typical visual artifacts in mosaic images are caused by differences in scene illumination and geometric misalignment. In Figure 4.1 these types of artifacts are illustrated. In what follows, the performance of the stitching algorithm proposed in chapter 3 is evaluated on stitching images with differences in illumination and geometric misalignment.

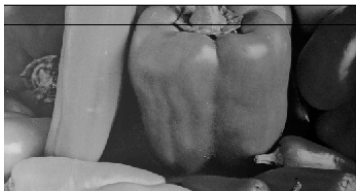
Analysis of the effects of photometric inconsistencies

In this section the performance of the method presented in 3.3.1 is illustrated with experiments. Given two partially overlapping, previously registered images with photometric inconsistencies (i.e., with differences in light intensity within the overlap region) the objective is to stitch them and produce a larger mosaic image which looks smooth and continuous without any noticeable artifacts such as seams or blurring.

The input images used to obtain the mosaic image were constructed from standard test images, by cropping the images in two and modifying the intensity values to generate photometric inconsistency.

In Figures 4.2a-4.2b an example of the input images used is shown. The two images have an overlapping region indicated by the two horizontal black lines. The intensity levels in the two images have been modified and are clearly different: the average

(a) Input image 1



(b) Input image 2

(c) Direct paste of intensities

(d) Mosaic using the proposed method

Figure 4.2: Results for test image `peppers`

intensity of the top image becomes lighter from bottom to top, while the intensity in the lower image becomes darker from left to right. This has the effect that in a "direct paste" mosaic image, the seam is less visible where the two image intensities are close (left hand-side of the image shown in Figure 4.2c, along the central horizontal line) and more visible where the difference in intensity is larger (right hand-side of Figure 4.2c). The mosaic image obtained using the method proposed in section 3.3.1 can be seen in Figure 4.2d. Clearly there is no seam and the image has no visual artifacts.

To illustrate the effect of Poisson smoothing at each resolution level during the Haar synthesis step of the image reconstruction from the stitched gradient, consider Figure 4.3. Reconstructing the image from the stitched gradient using the Haar wavelet approach [24] leads to blocking artifacts as shown in Figure 4.3a. These blocking artifacts are due to the fact that the stitched gradient is not a conservative

(a) Direct reconstruction from stitched gradient

(b) Reconstruction with Poisson solver at full resolution

(c) Poisson solver at each resolution

Figure 4.3: The effect of Poisson smoothing during the reconstruction

field and therefore there is a difference between the gradient of the reconstructed image and the stitched gradient. This difference can be reduced, and the visual appearance of the image improved, by using the reconstructed image in Figure 4.3a as the input to an iterative Poisson smoother at full resolution, as described in section 2.3.2. This leads, after several iterations, to Figure 4.3b with reduced but still visible blocking artifacts. Using the method of Hampton et al. [23] reviewed in section 2.3.2, with the Poisson solver at each resolution level during the synthesis step, as proposed in section 3.3.1, produces the image in Figure 4.3c. It can be seen that there are no blocking artifacts in this image.

To analyze the influence of including the Poisson solver during the reconstruction stage, the following experiment was conducted. The level of photometric inconsistency between the two input images was controlled by using a parameter k. Specifically,

two partially overlapping input images Φ_{upper} and Φ_{lower} of height n and width m were created from an image Φ, as described by the equations below:

for all $y \in \{1, 2, ..., n\}$ and $i \in \{1, 2, ..., n\}$

$$\Phi_{upper}(x, y) = \Phi(x, y) + (n - i)k, \text{ where } x \in \{1, 2, ..., m\} \tag{4.1}$$

for all $x \in \{1, 2, ..., m\}$ and $i \in \{0, 2, ..., m - 1\}$

$$\Phi_{lower}(x, y) = \Phi(x, y) - ik, \text{ where } y \in \{1, 2, ..., n\} \tag{4.2}$$

In this way, the intensity of the top image was increased upwards and the intensity of the bottom image was decreased from left to right, gradually, with respect to the corresponding regions from the original image.

The gradients of the two images were stitched as described in section 3.3.1, and the final mosaic image was obtained in three different ways. First, the image was reconstructed directly from the stitched gradient. Then, 50 iterations of the Poisson solver described by equation 2.40 were applied on the reconstructed image. Lastly, the Poisson solver described by equation 2.40 was included in the reconstruction stage by performing one iteration at each resolution level, as proposed in [23] and described in section 2.3.2 (see discussion in subsection "Obtaining an image from non-conservative vector fields"). The difference between the stitched gradient and the

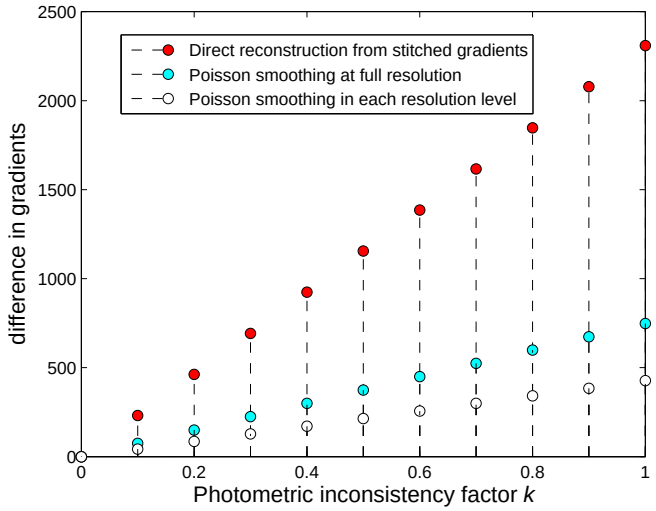
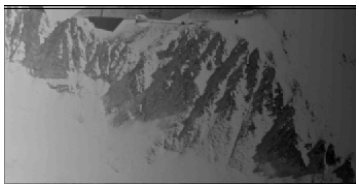


Figure 4.4: ℓ_2 norm of the difference between the stitched gradient and the gradient of the mosaic image. Mosaic size: 512×512; overlap region height: 100 pixels.

(a) Input image 1



(b) Input image 2

Figure 4.5: Input images for test image `jet`

(a) Direct paste of intensities

(b) Mosaic using the proposed method

Figure 4.6: Results for test image `jet`

gradient of the mosaic image reconstructed from the stitched gradient was evaluated for various values of k (see equations 4.1 and 4.2). The ℓ_2 norm of the difference between the stitched gradient and the gradient of the mosaic image reconstructed from the stitched gradient is plotted in Figure 4.4. It can be seen that the use of the Poisson solver at each resolution level leads to the smallest difference among the three options. In this case, this is consistent with the visual evaluation.

In our simulations, various sizes of the overlap region led to comparable results. For a 512×512 mosaic image, the smallest overlap region we considered was 6 pixels wide. The input images are shown in Figure 4.5. The results are still acceptable and an example is shown in Figure 4.6.

Figure 4.7 illustrates the performance of the proposed method with other types of illumination inconsistencies.

(a) Direct paste of intensities

(b) Mosaic using the proposed method

Figure 4.7: Results for test image `lena`

(a) Input images superimposed on the overlap region

(b) Direct paste of intensities

(c) Mosaic - misregistration example

Figure 4.8: Performance of stitching algorithm in case of misregistration

In Figure 4.7, the intensity of the upper image was obtained by gradually making the image darker towards its borders.

The overall quality of image mosaics is evaluated visually because ℓ_2 based measures may not yield meaningful results in the context of stitching images with differences in illumination. In particular, ℓ_2 based measures do not generally detect errors such as seams generated by stitching two images with very large photometric inconsistencies. To understand this, consider Figure 4.2c. As human vision is not sensitive to absolute illumination values, but rather to local contrast, the fact that we see two images with different mean intensity value (e.g., upper and lower halves of the image in Figure 4.2c) is not the problem. The problem in this image is the central horizontal seam, and the presence (or magnitude) of this seam cannot be correlated in a meaningful way with an ℓ_2 measure.

Analysis of the effects of geometric misalignment

In the previous section, the assumption was made that the images are perfectly registered. An interesting question is how possible registration errors between input images affect the quality of the mosaic image. Such errors may cause the appearance of ghosting artifacts in some methods. Figure 4.8 shows the stitching results when there is a misregistration between the input images of 10 pixels in both directions. The registration error can be observed in Figure 4.8b. The stitched image using the proposed method, Figure 4.8c, indicates that the proposed method can yield satisfactory results even in the presence of registration errors.

4.1.2 Stitching of grayscale images with irregularly shaped overlap region

In section 3.3.2, an algorithm was proposed to address the case when the images to be stitched have an irregularly shaped overlap region, after the registration. The aerial images shown in Figure 4.9a and 4.9b represent a typical example of this situation.

The main difficulties that the stitching algorithm must address in this situation is that the two images are of different orientations and scales. To increase the level of difficulty of the problem, the light intensity of the two images were modified before the method was tested. The result is shown in Figure 4.9d. In Figure 4.9c, for comparison purposes, the result obtained by stitching directly the luminance information is displayed. Clearly the method operating on the image gradients yields a natural

(a) Input image 1

(b) Input image 2

(c) Intensity stitching

(d) Gradient stitching

Figure 4.9: Stitching images with arbitrary shaped overlap: aerial example

looking mosaic image, without noticeable artifacts such as seams or other input image distortion.

4.1.3 Stitching of color images

Combining color pictures to create a panoramic image is a thorough studied problem in digital photography. Various algorithms have been proposed to date and implemented either in processing software available on computers (such as Adobe Photoshop, Hugin) or directly in the digital cameras. While camera solutions are effective, the tendency is towards an automatic approach, with as little user control as possible. In particular, a good algorithm should be able to automatically identify

(a) Input image 1

(b) Input image 2

Figure 4.10: Stitching of color images: beach landscape example - input images

(a) Averaging intensity

(b) Result using the proposed method

Figure 4.11: Stitching of color images: beach landscape example - result

the region of overlap between the input images and determine the feature points, without the intervention of the user. User intervention may allow for better results, to the expense of processing time. For example, manually selecting the feature points in the two images is often a tedious process and error prone. The only user controlled parameters in the registration method employed herein are two scaling factors. Aside from this, the proposed algorithm is automatic. The stitching technique introduced in section 3.3.3 was tested on a set of images, and the results obtained are analyzed herein.

Example 1: Creating a mosaic image from two images with different scales, and illumination conditions.

In Figure 4.10, the first pair of images on which the performance of the algorithm was evaluated is shown. The images were obtained by cropping overlapping parts of a

digital photography. To increase the level of difficulty of the stitching problem, in this case, the following modifications were performed on the images, before executing the algorithm. First, the resolution was modified, resulting in a scale difference between the two images. Then, the color in one of the images was changed, to simulate different illumination conditions. The result of the proposed technique is presented in Figure 4.11b. For comparison purposes, the result of averaging intensity values on the overlap region and cut and paste outside is displayed in Figure 4.11a. As can be seen, in the image processed exclusively in the intensity domain, visible seams mark the zones of overlap between the component images, whereas in the mosaic obtained with the proposed method, the transition from one image to the other is seamless.

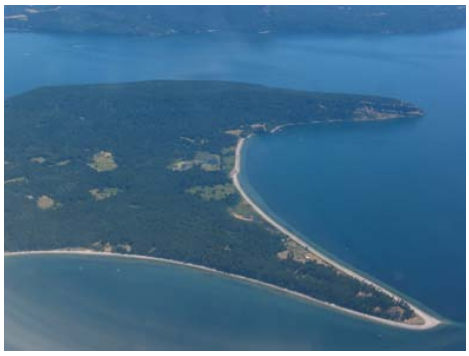
Example 2: Creating a mosaic image from two images with different scales, orientation and illumination conditions.

The next pair of images on which the proposed algorithm was tested can be seen in Figure 4.12.

The challenges posed by this set of images are numerous. The set of images are real images, taken with a point and shoot digital camera from the window of an airplane. As a result, there is a perspective type distortion between the two scenes, as well as a difference in orientation, scale and illumination (the image in Figure 4.12b is brighter than the image in Figure 4.12a). Another consequence of this image acquisition method are the halo shaped glass reflections that can be seen in the lower part of both images. Lastly, the images are blurred due to motion (of the camera and of plane) and the color is faded, due to filtering by the thick glass of the window. None of the artifacts present in the image at its acquisition was eliminated prior stitching. The result of the proposed method for the aerial images is shown in Figure 4.13b and as can be seen, the proposed method works well and produces a reasonable result even in the case of severely distorted input images.

Example 3: Creating a mosaic images from three input images.

In general, a panoramic view of a scene is obtained from more than two representations of the imaged scene. The algorithm, as described in sections 3.3.2 and 3.3.3, considers the case of stitching two images with a common region, to generate a larger representation of the imaged scene. The following example illustrates the fact that the proposed technique can be extended in a more general setup, to produce seamless



(a) Input image 1

(b) Input image 2

Figure 4.12: Stitching of color images: aerial example - input images

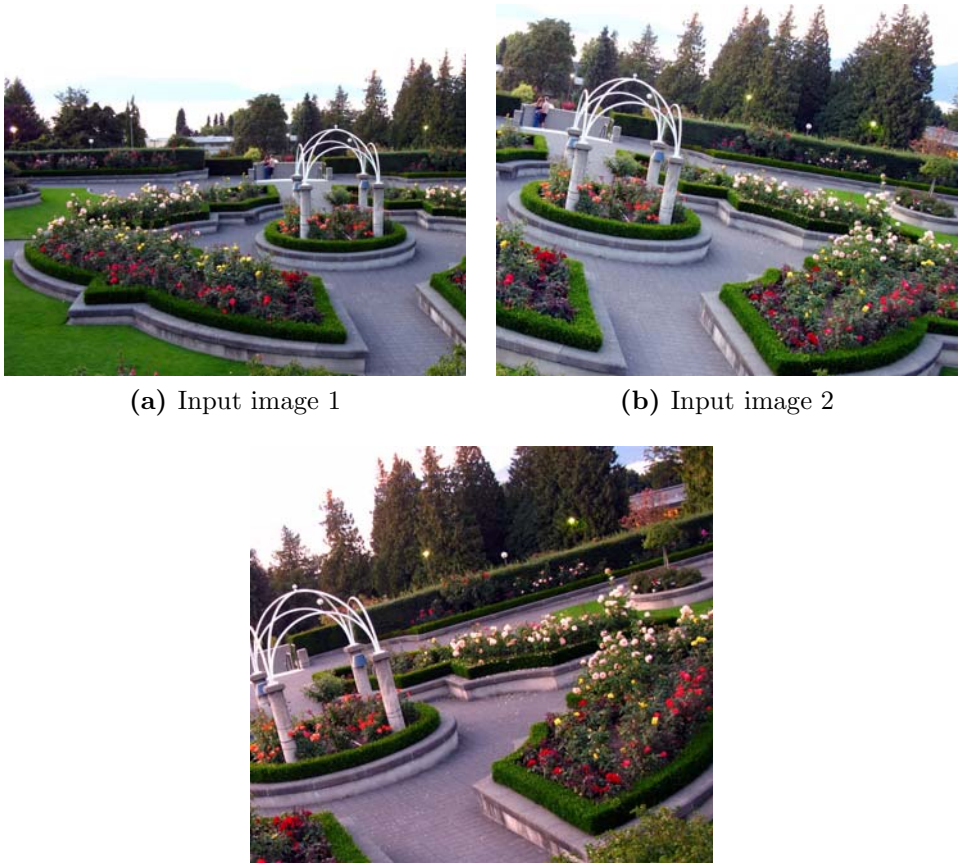
(a) Averaging intensity

(b) Result using the proposed method

Figure 4.13: Stitching of color images: aerial example - result

mosaics from more than two input images.

In Figure 4.14 three parts of a garden scenery are shown. This set of testing images was obtained from a digital photography, by first cropping three overlapping parts. To simulate typical problems that pose difficulties to stitching algorithms in practice, and increase the level of difficulty of the stitching problem, the following



(a) Input image 1 **(b)** Input image 2

(c) Input image 3

Figure 4.14: Stitching of color images: garden example - input images

modifications were performed on the initial set of overlapping image parts. First, the resolution was modified, resulting in scaling differences between the three images. Next, the second input image was rotated by 10 ° counterclockwise and its illumination level was increased. Then, the third image was rotated by 15 ° clockwise and the information contained in the red channel was significantly increased.

The approach of the stitching algorithm to combine the three images consists of two main stages. First, the two input images shown in Figure 4.14a and 4.14b) are registered, as shown in Figure 4.15a and then stitched together. This initial result is displayed in Figure 4.15b.

Next, the objective is to stitch the intermediate result displayed in Figure 4.15b with the third input image, shown in Figure 4.15c.

Directly combining the composite image obtained from the first two images with the third image as described in section 3.3.2 generates an error in the gradient of the

Table 4.1: RGB average values of input images versus mosaic image

Image	R	G	B	R : G : B ratio
Image 1	104	110	103	0.41 : 0.43 : 0.40
Image 2	110	113	109	0.43 : 0.44 : 0.43
Image 3	101	90	86	0.40 : 0.35 : 0.34
Mosaic	103	97	93	0.40 : 0.38 : 0.36

final mosaic, essentially due to the region of support of the composite image obtained after the first two images have been stitched. This shortcoming can be eliminated by a simple modification of the method and its software implementation. Another approach to overcome this shortcoming and create a mosaic from the two images using the current algorithm description and implementation is to first crop the composite image obtained from the first two images to the smallest fully supported rectangular image and stitch this with the third image. The concept of "fully supported image" is used here to designate an image without additional padding content. For instance, the third input image (Figure 4.15c) is fully supported, whereas the intermediate image (Figure 4.15b) is not.

The result obtained by stitching the cropped intermediate image with the third image is shown in Figure 4.16. As can be seen, the mosaic image displayed in Figure 4.16 has no visible seams and it is visually plausible. An interesting observation concerns the mean value of this image. As can be seen, the final mosaic is more similar in aspect to the third input image, namely, it seems to have a high value of red channel information. This is, indeed, the case, as can be seen from Table 4.1, where the average intensity values of each channels of the reconstruction are displayed, together with the corresponding values of the three input images. The values displayed in the last column represent the ratio of the three corresponding RGB values, normalized to 255. The higher average intensity value in the red channel of the mosaic image is a direct consequence of the way in which the mean value of the composite image is estimated to reconstruct the mosaic from the gradient data set (see section 3.3.3 for details). In situations when this estimate does not produce the desired result, namely, a natural looking image, the average value of a particular channel can be corrected to produce a more natural looking image, as is illustrated in Figure 4.17. Here, the mean value in the red channel was lowered.

The source images stitched in this section were also combined using the Photomerge tool from Adobe Photoshop Extended CS5. The blending part of this tool

(a) Registering the first two images

(b) Result of stitching the first two images

(c) Input image 3

Figure 4.15: Stitching of color images: garden example

Figure 4.16: Stitching of color images: garden example - result.

Figure 4.17: Stitching of color images: mean value corrected in one channel

was developed by A. Agarwala [1] and operates in the gradient domain. The composite image is recovered from the gradient by an efficient Poisson solver. The software produced very similar results for all tested datasets, with the exception of the two aerial images (Fig. 4.12), for which the automatic registration part of Photomerge failed.

4.2 Seamless object insertion

In section 3.4, a conceptually simple object insertion method which operates in the gradient domain was presented. In this section, the performance of this method is illustrated with examples. The results show that performing the same operation in the gradient domain produces better results than operating directly in the intensity domain, reinforcing the idea that in image processing applications such as the ones analyzed here, the gradient domain provides a more adequate framework than the intensity domain. In Figure 4.18, the two images to be combined are presented, with the regions of interest highlighted in each by a black rectangle. In Figure 4.19a, the result obtained by a "cut and paste" approach applied directly in the intensity domain is shown. The transition from one image to the other is clearly visible along the horizontal line that crosses the composite image. In Figure 4.19b, the result obtained using the method presented in section 3.4 can be seen. The transition from one image to the other is barely noticeable and the image has a natural aspect.

(a) Source image **(b)** Target image

Figure 4.18: Seamless object insertion: swim example - input images.

(a) Pasting in intensity domain

(b) Seamless object insertion in gradient domain

Figure 4.19: Seamless object insertion: swim example - result.

The determining factor in the quality of the result using this approach is choosing the region of interest. The method yields plausible results only if the textures and colors of the two images to be combined are similar around the boundary of the region of interest.

The following example illustrates the method in a slightly different setup. Given source and target image as shown in Figure 4.20, the objective is to insert the entire content of the source image into the region of interest marked by the black rectangle in the target image in 4.20b.

The approach in this case is to first rescale the source image to match the dimension of the region of interest in the target image. Then, the method as described in section 3.4 is applied and the result obtained is displayed in Figure 4.21b. For comparison purposes, in Figure 4.21a, the result obtained by a "cut and paste" operation in the intensity domain is shown. The four edges that mark the transition between the source image content and the target image content are clearly visible in the result obtained working in the intensity domain, whereas the same approach operating in the gradient domain leads to significantly improved results.

In this work the object insertion method was introduced in the context of image morphing. Therefore, in what follows an example illustrating the method in this particular context is presented. Given the two faces shown in Figures 4.22a and 4.22b, the objective is to combine features of the two individuals and produce a new face that looks natural.

(a) Source image **(b)** Target image

Figure 4.20: Seamless object insertion: books example - input images.

(a) Pasting in intensity domain **(b)** Seamless object insertion in gradient domain

Figure 4.21: Seamless object insertion: books example - result.

The two input images labelled Face A and Face B in Figure 4.22 are from the Yale Face Database [18]. The images have the same dimension and represent portraits of two individuals taken under similar illumination conditions.

The approach in this experiment was conducted was sequential. First, a region of interest was identified in Face B by specifying the coordinates of the location of the feature to be moved in the target image Face A (namely, the eyes region). Then, this region of interest was inserted in the corresponding location from Face A. Face C, shown in Figure 4.22c, is the result obtained when the insertion is done in intensity domain, and Face D, shown in Figure 4.22d is the result obtained using the method presented in section 3.4.

Next, the region of interest of the second feature (namely, the mouth region) was identified in Face B by specifying its coordinates. This region of interest was then inserted in Faces C and D. Face E is obtained by "cut and paste" of intensity values,

(a) Face A

(b) Face B

(c) Face C (face A with eyes from face B)

(d) Face D (face A with eyes from face B)

(e) Face E (face C with mouth from face B)

(f) Face F (face D with mouth from face B)

Figure 4.22: Image morphing: face example. Top row: input images. Middle and third row: composite images; left-hand side: "cut and paste" of intensities, right-hand side: the results using the proposed gradient based method.

whereas face F is obtained by operating in the gradient domain and reconstructing the image as described in section 3.4. As can be seen from Figure 4.22f, the gradient based method produced a natural looking composite face, uniting features from Face A and Face B.

The experiment could have been conducted in a single stage, by moving both features of interest of Face B to Face A. However, a sequential approach was preferred in this case to illustrate two different ways in which the features of Face A and B can be combined to produce two different equally natural looking faces (namely, Face D and Face F).

This example was presented to stress the fact that the simple object insertion method presented in section 3.4 can yield good results if the images to be combined have similar illumination conditions and similar textures.

4.3 Performance evaluation

The quality of the results presented in this chapter was assessed subjectively. A general agreement in literature [10, 38] seems to indicate that subjective evaluation prevails in assessing the quality of composite images produced by various algorithms. In the two applications studied in this thesis, subjective evaluation begins by comparing the resulting image with the input images. This visual comparison must focus on the resemblance between the composite image and the input images in the non-overlapping parts, and on the seamless, plausible transition from one input image to the other. Although it seems a straightforward task, a good subjective evaluation of images is difficult to perform in practice. Specifically, there are several critical conditions that must be met in order for the subjective evaluation study results to be reliable. First, the evaluation process should be conducted by a sufficiently large group of evaluators. Then, all monitors of the evaluators should be calibrated in the same way and the ambient light conditions during the study of the images should be the same, to allow for a fair comparison. In the absence of these factors, subjective evaluation can easily lead to inaccurate results.

Based on this discussion, it follows that an objective measure to evaluate the quality of the composite images obtained in sections 4.1 and 4.2, is difficult to define. Boutellier et al. [9] attempted to develop an objective measure of evaluating the quality of image mosaics. In their work, the mosaics are produced from a set of images obtained from a given image, stitched together through several algorithms,

and the results are compared with the original image, after the two images are again registered, to allow for pixelwise comparisons. The authors claim that the evaluation method produces reliable results in detecting blurring artifacts or edge discontinuities in the mosaic image, but does not account for changes in light intensity. However, as was discussed in section 4.1.1, a visually satisfactory stitched image may have different intensities than the input images, which will produce unsatisfactory results for this objective measure. On the other hand, if the intensity values in the composite image are similar to the intensity values in the input images, but there is a visible seam in the image, this may result in an acceptable value of this objective measure, but uncorrelated with the subjective evaluation of the image. Therefore, this method may not yield meaningful results in detecting several types of artifacts.

A possible approach to evaluate the performance of the presented methods is by evaluating the performance of the algorithms used at each stage, such as registration and reconstruction from gradient. Although this can be done by using objective measures, it will not necessarily give an indication of the visual quality of the final result. The registration algorithm employed in the proposed stitching technique is presented and evaluated in [49, 50] and shown to be robust to affine distortions, differences in scale and orientation and noise corruption in the input images. These conclusions were also confirmed by the results presented in section 4.1. The problem of image reconstruction from gradient was investigated in detail in chapter 2. A good reconstruction method should be robust to the presence of errors and/ or noise in the given gradient. Also, the increase in size of the testing data set should not pose a problem for the reconstruction technique. In section 2.4.1, a comparative study was performed to illustrate the performance of the Haar wavelet based reconstruction algorithm employed in the proposed stitching and object insertion algorithms. A comparison between this reconstruction technique and another Poisson-based numerical integration techniques is also presented in [23].

4.4 Conclusions

In this chapter, two applications of image processing in the gradient domain have been illustrated with examples. The experimental results illustrate the methods presented in chapter 3 and show that they yield seamless composite images despite luminance or color differences between the input images. Finally, a discussion regarding the assessment of the quality of the results produced using the two methods was presented.

Chapter 5

Conclusions

5.1 Conclusions

In this thesis, the focus was on studying the performance of gradient domain based algorithms in the context of two image processing applications. In particular, the problems of image stitching and object insertion were considered and two methods providing a solution were presented. The proposed methods include conceptually simple gradient domain manipulations and rely on recent progress in the area of image registration and image reconstruction from gradient.

The work reported in this thesis can be summarized as follows.

In chapter 2, the problem of image reconstruction from gradient was presented, as a necessary final stage in image processing techniques operating in the gradient domain. Several approaches to recover the image from a given gradient data set were reviewed. In particular, the focus was on presenting the reconstruction method of Hampton et al. [23, 24]. An improvement was proposed to this reconstruction algorithm. Experimental results showed that this modification of the method produces a technique that perfectly recovers the image from noise free gradient data.

The main contribution of chapter 2 relies in conducting a comparison of four reconstruction algorithms, using as evaluation criteria the accuracy of the solutions they produce and their computational speed. The comparison study was conducted in two scenarios, depending on the amount of noise or errors corrupting the given gradient from which the image was reconstructed. In the case of recovering the image from a noise free gradient, analysis showed that the reconstruction method of Hampton et al. [24], with the elimination of the waffle pattern proposed in this thesis,

yields the best results in terms of solution accuracy in a comparable time with the fastest of the four analyzed algorithms. In the case of recovering the image from noise corrupted gradient data, we distinguished the following situations. For medium values of SNR in the input gradient, the best performance was obtained with a DCT-based Poisson solver [37], and close results were achieved with the method of Hampton et al. [23], with a Poisson solver included during synthesis. For extreme situations of very low values of SNR in the input gradient, the best performance was given by a DFT based algorithm [16], followed by the DCT based Poisson solver [37] and then by the Hampton et al. method [23].

In chapter 3, two image processing problems and two methods for their treatment were introduced. First, the problem of image stitching was presented and a gradient based stitching algorithm was developed, starting from the method proposed in [36]. The proposed stitching algorithm has three main stages: wavelet based image registration, gradient based blending and image reconstruction from gradient, and was developed to addresses situations of gradually increasing difficulty and higher levels of generality. In the remainder of chapter 3, the problem of object insertion was described in the context of image morphing and a conceptually simple object insertion method was presented.

The image stitching and object insertion methods proposed in this thesis are gradient based. Therefore, an essential part of both of these techniques is recovering the final image from a given gradient data set. Based on the analysis in chapter 2, the reconstruction method of Hampton et al. [23] was selected for this purpose and employed to produce the resulting image in the proposed stitching and object insertion techniques.

The quality of the results produced by the proposed image stitching and object insertion methods was subjectively evaluated on various examples presented in chapter 4. A general discussion concerning the evaluation of such algorithms was provided at the end of the chapter. In light of this discussion and of the subjective evaluation of the results in this chapter, it was concluded that the techniques proposed in this thesis are a good approach to the considered applications.

5.2 Future directions

Several problems related to the work described in this thesis appear to be worthwhile to pursue. In particular, efforts can be made to further investigate and improve the

main stages of the stitching method proposed in section 3.3, namely the registration and the reconstruction algorithms.

The registration algorithm employed in the proposed stitching method is wavelet based and operates on the intensity domain of the images to be aligned. Consequently, the algorithm is sensitive to significant illumination differences between the images. It seems worthwhile investigating performing the registration part of the stitching algorithm in the gradient domain of the images to be combined, as this may result in an overall improvement of the results obtained when there are significant illumination differences.

Obtaining a fully automatic stitching method given a set of images is another objective subject of our research interest. The registration stage from the proposed stitching technique requires two input parameters, in order to account for existing scale differences between the two images. In [19] and in this work, choosing these parameters is done empirically. Our belief is that these scaling parameters are embedded in the structure of the images - in their intensity or their gradient, and efforts towards an automatic identification of these parameters would definitely improve the proposed method.

The reconstruction algorithm of Hampton et al. is used in this thesis in the context of image stitching and morphing by object insertion. The good results obtained here seem to indicate that the method could be used as a final stage to reconstruct images from gradient measurements in other gradient based applications. In particular, applications such as transparency or high dynamic range imaging seem worthwhile to further analyze.

Another direction in which the reconstruction algorithm could be developed is that of denoising by estimating the noise level from the high frequency band quadrant of the decomposition obtained after the analysis step. A thresholding policy [12] employed before the reconstruction may improve the quality of the results in the presence of noise in the given gradient.

Bibliography

[1] A. Agarwala. Efficient gradient-domain compositing using quadtrees. *ACM Transactions on Graphics*, 26, Jul. 2007.

[2] A. Agarwala, M. Dontcheva, M. Agrawala, S. Drucker, A. Colburn, B. Curless, D. Salesin, and M. Cohen. Interactive digital photomontage. *ACM Transactions on Graphics*, 23(3):294–302, Aug. 2004.

[3] A. Agrawal. Research website. www.umiacs.umd.edu/~aagrawal/.

[4] A. Agrawal, R. Chellappa, and R. Raskar. An algebraic approach to surface reconstruction from gradient fields. In *Tenth IEEE International Conference on Computer Vision (ICCV'05)*, volume 1, pages 174–181. IEEE, 2005.

[5] A. Agrawal, R. Raskar, and R. Chellappa. What is the range of surface reconstructions from a gradient field? In A. Leonardis, H. Bischof, and A. Pinz, editors, *Computer Vision a ECCV 2006*, volume 3951 of *Lecture Notes in Computer Science*, pages 578–591. Springer Berlin, Heidelberg, 2006.

[6] A. Agrawal, R. Raskar, S. K. Nayar, and Y. Li. Removing photography artifacts using gradient projection and flash-exposure sampling. *ACM Transactions on Graphics*, 24(3):828–835, Jul. 2005.

[7] A. Antoniou and W.-S. Lu. *Practical Optimization: Algorithms and Engineering Applications*. Springer, New York, USA, 2007.

[8] A. Ardeshir Goshtasby. *2-D and 3-D Image Registration: for Medical, Remote Sensing, and Industrial Applications*. John Wiley & Sons Inc., Hoboken, NJ, USA.

[9] J. Boutellier, O. Silvén, M. Tico, and L. Korhonen. Objective evaluation of image mosaics. In J. Braz et al., editor, *Computer Vision and Computer Graphics. Theory and Applications*, pages 107–117. Springer Berlin, Heidelberg, 2008.

[10] P.J. Burt and E.H. Adelson. A multiresolution spline with application to image mosaics. *ACM Transactions on Graphics*, 2:217–236, Oct. 1983.

[11] A. Criminisi, P. Perez, and K. Toyama. Region filling and object removal by exemplar-based image inpainting. *IEEE Transactions on Image Processing*, 13:1200–1212, Sept. 2004.

[12] D.L. Donoho. De-noising by soft-thresholding. *IEEE Transactions Information Theory*, 41(3):613–627, may 1995.

[13] R. Fattal, D. Lischinski, and M. Werman. Gradient domain high dynamic range compression. *ACM Transactions on Graphics*, 21(3):249–256, Jul. 2002.

[14] G.D. Finlayson, Hordley. S.D., and M.S. Drew. Removing shadows from images. In *In ECCV 2002: European Conference on Computer Vision*, pages 823–836, 2002.

[15] J. Foley and A. van Dam. *Fundamentals of Interactive Computer Graphics*. Addison-Wesley Publishing Company, MA, USA, 1982.

[16] R.T. Frankot and R. Chellappa. A method for enforcing integrability in shape from shading algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10:439–451, Jul. 1988.

[17] D.L. Fried. Least square fitting a wavefront distortion estimate to an array of phase difference measurements. *Journal of the Optical Society of America*, 67:370–374, 1977.

[18] A.S. Georghiades, P.N. Belhumeur, and D.J. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Trans. Pattern Anal. Mach. Intelligence*, 23(6):643–660, 2001.

[19] S. Gillan. A technique for face recognition based on image registration. Master's thesis, University of Victoria, 2009.

[20] S. Gillan, P. Agathoklis, and M.S. Yasein. A feature based technique for face recognition using Mexican hat wavelets. In *IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*, pages 792–797, Aug. 2009.

[21] L. Gottesfeld Brown. A survey of image registration techniques. *ACM Computing Surveys (CSUR)*, 24(4):325–376, Dec. 1992.

[22] P. J. Hampton. *Robust order N wavelet filterbanks to perform 2-D numerical integration directly from partial difference or gradient measurements*. PhD thesis, University of Victoria, 2009.

[23] P.J. Hampton and P. Agathoklis. Comparison of Haar wavelet-based and Poisson-based numerical integration techniques. In *Proceedings of IEEE International Symposium on Circuits and Systems*, pages 1623–1626, 2010.

[24] P.J. Hampton, P. Agathoklis, and C. Bradley. A new wave-front reconstruction method for adaptive optics system using wavelets. *IEEE Journal of Selected Topics in Signal Processing*, 2(5):781–792, Oct. 2008.

[25] P.J. Hampton, P. Agathoklis, and C. Bradley. Wavefront reconstruction over a circular aperture using gradient data extrapolated via the mirror equations. *Applied Optics*, 48(20):4018–4030, Jul. 2009.

[26] C.-T. Hsu and J.-L. Wu Wu. Multiresolution mosaic. *IEEE Transactions on Consumer Electronics*, 42(4):981–990, Nov. 1996.

[27] R. Hudgin. Wave-front reconstruction for compensated imaging. *Journal of the Optical Society of America*, 67:375–378, 1977.

[28] J. Jia and C.-K. Tang. Eliminating structure and intensity misalignment in image stitching. In *Tenth IEEE International Conference on Computer Vision (ICCV'05)*, volume 2, pages 1651–1658. IEEE, 2005.

[29] C. Kuhn. Chuck kuhn photography. www.pbase.com/ckuhn55/.

[30] A. Levin, A. Zomet, S. Peleg, and Y. Weiss. Seamless image stitching in the gradient domain. In *Eighth European Conference on Computer Vision*, volume 4, pages 377–389, 2004.

[31] P. Perez, M. Gangnet, and A. Blake. Poisson image editing. In *Proceedings of SIGGRAPH*, pages 313–318, Jul. 2003.

[32] L. A. Poyneer, D. T. Gavel, and J. M. Brase. Fast wave-front reconstruction in large adaptive optics systems with use of the Fourier transform. *Journal of the Optical Society of America*, 19(10):2100–2111, Oct. 2002.

[33] D. Reddy, A. Agrawal, and R. Chellappa. Enforcing integrability by error correction using ℓ_1-minimization.

[34] M. Schenk. The machband effect. www.markschenk.com/various/machband.html.

[35] S. M. Seitz and C. R. Dyer. View morphing. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, SIGGRAPH '96, pages 21–30, New York, NY, USA, 1996. ACM.

[36] I.S. Sevcenco, P.J. Hampton, and P. Agathoklis. Seamless stitching of images based on a Haar wavelet 2D integration method. In *17th International Conference on Digital Signal Processing (DSP)*, pages 1–6, Jul. 2011.

[37] T. Simchony, R. Chellappa, and M. Shao. Direct analytical methods for solving Poisson equations in computer vision problems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 435–446, May. 1990.

[38] M.-S. Su, W.-L. Hwang, and K.-Y. Cheng. Analysis on multiresolution mosaic images. *IEEE Transactions on Image Processing*, 13(7):952–959, Jul. 2004.

[39] R. Szeliski. Video mosaics for virtual environments. *IEEE Computer Graphics and Applications*, pages 22–30, Mar. 1996.

[40] R. Szeliski. Image alignment and stitching: a tutorial. Technical Report MSR-TR-2004-92, Microsoft Research, 2004.

[41] J. Tumblin, A. Agarwal, and A. Raskar. Why I want a gradient camera. In *Proceeding of IEEE Computer Society Conf. Computer Vision and Pattern Recognition*, volume 1, pages 103–110, 2005.

[42] M. Uyttendaele, A. Eden, and R. Skeliski. Eliminating ghosting and exposure artifacts in image mosaics. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 509–516, 2001.

[43] P. P. Vaidyanathan. *Multirate Systems and Filter Banks*. Prentice-Hall, Englewood Cliffs, NJ, USA, 1993.

[44] M. Vetterli and J. Kovačević. *Wavelets and Subband Coding*. Prentice-Hall, Englewood Cliffs, NJ, USA, 1995.

[45] Z. Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, Apr. 2004.

[46] D. S. Watkins. *Fundamentals of Matrix Computations*. John Wiley & Sons, Inc., New York, USA, second edition, 2002.

[47] J. J. Widiker, S. R. Harris, and B. D. Duncan. High-speed Shack-Hartmann wavefront sensor design with commercial off-the-shelf optics. *Applied Optics*, 45(2):383–395, Jan. 2006.

[48] G. Wolberg. Recent advances in image morphing. In *Proceedings of the 1996 Conference on Computer Graphics International*, pages 64–71, 1996.

[49] M.S. Yasein and P. Agathoklis. A robust, feature-based algorithm for aerial image registration. In *IEEE International Symposium on Industrial Electronics*, pages 1731–1736, June 2007.

[50] M.S. Yasein and P. Agathoklis. A feature-based image registration technique for images of different scale. In *IEEE International Symposium on Circuits and Systems*, pages 3558–3561, May 2008.

Appendix A

Additional Information

Table A.1: Reconstruction from noisy gradient - test image `ramp peaks`

SNR in gradient	Relative error re			
	DCT Poisson solver [37]	Frankot-Chellappa [16]	Method from [24]	Method from [23]
12.4948	0.0446	0.1460	0.0515	0.0490
6.7726	0.1050	0.1533	0.1159	0.1058
3.1423	0.1370	0.1933	0.1550	0.1376
0.6593	0.1836	0.2068	0.2073	0.1926
-1.3178	0.1890	0.2169	0.2528	0.2124
-2.8539	0.2876	0.2359	0.3224	0.3021
-4.4620	0.3466	0.2944	0.3910	0.3652
-5.3237	0.3704	0.3163	0.4275	0.3982
-6.4405	0.4610	0.3960	0.5112	0.4780

Table A.2: Reconstruction from noisy gradient - test image `airplane`

SNR in gradient	Relative error re			
	DCT Poisson solver [37]	Frankot-Chellappa [16]	Method from [24]	Method from [23]
12.5186	0.0260	0.1063	0.0314	0.0287
6.5476	0.0599	0.1188	0.0699	0.0630
2.9923	0.0749	0.1263	0.0895	0.0804
0.4638	0.1059	0.1395	0.1273	0.1096
-1.3990	0.1687	0.1580	0.1862	0.1710
-2.9532	0.1713	0.1713	0.1964	0.1762
-4.4436	0.1843	0.1920	0.2231	0.1930
-5.5486	0.2024	0.2048	0.2494	0.2083
-6.5509	0.2879	0.2282	0.3190	0.2939

Table A.3: Reconstruction from noisy gradient - test image `camera`

SNR	Relative error re			
in gradient	DCT Poisson solver [37]	Frankot-Chellappa [16]	Method from [24]	Method from [23]
12.6716	0.0383	0.1862	0.0457	0.0418
6.6641	0.0772	0.1884	0.0891	0.0812
3.1484	0.1253	0.2015	0.1448	0.1319
0.6554	0.1403	0.2020	0.1712	0.1502
-1.3084	0.1772	0.2338	0.2145	0.1860
-2.8729	0.2448	0.2868	0.2821	0.2493
-4.1928	0.3082	0.2956	0.3493	0.3206
-5.4020	0.3931	0.3030	0.4458	0.4045
-6.4097	0.3567	0.3322	0.4210	0.3716

Table A.4: Reconstruction from noisy gradient - test image `barbara`

SNR	Relative error re			
in gradient	DCT Poisson solver [37]	Frankot-Chellappa [16]	Method from [24]	Method from [23]
14.8260	0.0349	0.1502	0.0405	0.0377
8.8037	0.0669	0.1560	0.0787	0.0687
5.2844	0.0972	0.1717	0.1112	0.0980
2.7797	0.1312	0.1776	0.1553	0.1389
0.8447	0.1585	0.1903	0.1910	0.1647
-0.7433	0.1897	0.2314	0.2254	0.1944
-2.0855	0.2337	0.2385	0.2696	0.2363
-3.2225	0.2475	0.2725	0.3007	0.2604
-4.2730	0.2685	0.2734	0.3331	0.2802

Table A.5: Reconstruction from noisy gradient - test image `man`

SNR	Relative error re			
in gradient	DCT Poisson solver [37]	Frankot-Chellappa [16]	Method from [24]	Method from [23]
12.5173	0.0302	0.1999	0.0367	0.0333
6.4988	0.0597	0.2139	0.0735	0.0634
2.9743	0.0966	0.2087	0.1159	0.1008
0.4868	0.1189	0.2238	0.1512	0.1246
-1.4630	0.1422	0.2372	0.1787	0.1495
-3.0392	0.1935	0.2439	0.2243	0.1924
-4.3890	0.2236	0.2829	0.2663	0.2281
-5.5460	0.2275	0.2792	0.2865	0.2410
-6.5600	0.3003	0.3080	0.3563	0.3085

Table A.6: Image reconstruction from noisy gradient - centralizing chart obtained by averaging the relative errors of the reconstructions obtained from similar values of SNR in input gradient for all tested images

SNR in gradient	Relative error re			
	DCT Poisson solver [37]	Frankot-Chellappa [16]	Method from [24]	Method from [23]
12.5506	0.0348	0.1596	0.0413	0.0382
6.6208	0.0754	0.1686	0.0871	0.0784
3.0643	0.1084	0.1824	0.1263	0.1127
0.5663	0.1372	0.1931	0.1643	0.1443
-1.3720	0.1693	0.2115	0.2081	0.1797
-2.9298	0.2243	0.2345	0.2563	0.2300
-4.3719	0.2657	0.2662	0.3074	0.2767
-5.4551	0.2984	0.2758	0.3523	0.3130
-6.4903	0.3515	0.3161	0.4019	0.3630