Color Face Recognition using Quaternionic Gabor Filters

Creed F. Jones III
Bradley Department of Electrical and Computer Engineering
Virginia Polytechnic Institute and State University

Doctoral Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Computer Engineering

A. Lynn Abbott, Chair
Richard W. Conners
Roger W. Ehrich
Ira Jacobs
Scott Midkiff

December 13, 2004
Blacksburg, Virginia

Keywords: face recognition, color, Gabor filters, hypercomplex, quaternions

Copyright 2004, Creed F. Jones III
Color Face Recognition using Quaternionic Gabor Filters

Creed F. Jones III

Abstract

This dissertation reports the development of a technique for automated face recognition, using color images. One of the more powerful techniques for recognition of faces in monochromatic images has been extended to color by the use of hypercomplex numbers called quaternions. Two software implementations have been written of the new method and the analogous method for use on monochromatic images. Test results show that the new method is superior in accuracy to the analogous monochrome method.

Although color images are generally collected, the great majority of published research efforts and of commercially available systems use only the intensity features. This surprising fact provided motivation to the three thesis statements proposed in this dissertation.

The first is that the use of color information can increase face recognition accuracy. Face images contain many features, some of which are only easily distinguishable using color while others would seem more robust to illumination variation when color is considered.

The second thesis statement is that the currently popular technique of graph-based face analysis and matching of features extracted from application of a family of Gabor filters can be extended to use with color. A particular method of defining a filter appropriate for color images is used; the usual complex Gabor filter is adapted to the domain of quaternions. Four alternative approaches to the extension of complex Gabor filters to quaternions are defined and discussed; the most promising is selected and used as the basis for subsequent implementation and experimentation.

The third thesis statement is that statistical analysis can identify portions of the face image that are highly relevant – i.e., locations that are especially well suited for use in face recognition systems. Conventionally, the Gabor-based graph method extracts features at locations that are equally spaced, or perhaps selected manually on a non-uniform graph. We have defined a relevance image, in which the intensity values are computed from the intensity variance across
a number of images from different individuals and the mutual information between the pixel distributions of sets of images from different individuals and the same individual.

A complete software implementation of the new face recognition method has been developed. Feature vectors called jets are extracted by application of the novel quaternion Gabor filter, and matched against models of other faces. In order to test the validity of the thesis statements, a parallel software implementation of the conventional monochromatic Gabor graph method has been developed and side-by-side testing has been conducted. Testing results show accuracy increases of 3% to 17% in the new color-based method over the conventional monochromatic method. These testing results demonstrate that color information can indeed provide a significant increase in accuracy, that the extension of Gabor filters to color through the use of quaternions does give a viable feature set, and that the face landmarks chosen via statistical methods do have high relevance for face discrimination.
Acknowledgments

This dissertation would not exist without the assistance and support of many people. I thank my committee, Drs. Conners, Ehrich, Jacobs and Midkiff for agreeing to guide me through this process, and most especially Dr. A Lynn Abbott, my committee chair. His direction, advice and comments were essential, and I know that the time he devoted to my studies could have been used in many other ways; I thank him for the sacrifice.

Many colleagues in the biometrics industry provided comment, input, or even just interest. Sincere appreciation is extended to Drs. Paul Griffin, Colin Soutar and Don D'Amato for the benefit of their expertise. An extensive telephone conversation with Dr. John Daugman, whom I did not previously know, was crucial in my understanding not just what the two-dimensional Gabor filter was, but why it was that way and what it all meant; my thanks to him.

Sagem Morpho Inc. of Tacoma, WA, provided research funding for a great part of this work; their support is most appreciated. Denis Limongi and Jean-Christophe Fondeur were a great help in the early days of trying to put some reasonable bounds around the topic of face recognition. Special mention must be made of Glen McNeil at Sagem Morpho. His support, motivation and friendship through this process was more essential than he may realize.

My colleagues at Seattle Pacific University were essential for their assistance and support. Special thanks to my departmental colleagues – Professors Mike Tindall, Phil Prins, Elaine Weltz and Chuck Burris – and to Dr. Ed Smyth for counsel, prayer and friendship.

On a higher level, appreciation and recognition must go to my wife, Jana, and our daughter Laura. They were willing to endure the major life change involved in becoming graduate students again, so that I might study for the PhD. Once the studies began, they were always there to encourage me – which sometimes meant talking me out of despair! They both have my undying love.

Finally, study of the incredible task of recognizing people by their face has left me with awed respect for the One who designed us all to do such an amazing thing, effortlessly, day in and day out, in all sorts of conditions – to God is all glory.
# Table of Contents

Chapter 1  
Background & Motivation  
1.1 Introduction ........................................................................................................... 1  
1.2 The Need for Face Recognition ........................................................................... 1  
1.3 Current Face Recognition Technology ................................................................. 3  
  1.3.1 Background ........................................................................................................ 3  
  1.3.2 FERET/FRVT Tests ............................................................................................ 6  
1.4 Limitations of Computational Face Recognition Systems .................................. 9  
  1.4.1 Peak accuracy under ideal conditions ............................................................... 9  
  1.4.2 Pose - illumination – expression ...................................................................... 9  
  1.4.3 Aging ................................................................................................................ 10  
  1.4.4 Other Potential Sources of Variability ............................................................. 10  
1.5 Current use of color .............................................................................................. 11  
  1.5.1 Face Detection .................................................................................................. 11  
  1.5.2 Eye Location ..................................................................................................... 12  
  1.5.3 Extension of the Eigenface Method ................................................................. 12  
1.6 Thesis Statements ................................................................................................ 13  
  1.6.1 Use of Color Information Can Increase Accuracy .......................................... 13  
  1.6.2 Extending Gabor Techniques to Color Can Increase Accuracy .................. 14  
  1.6.3 Statistical Methods Can Improve Face Landmark Location ......................... 14  
1.7 Contributions of This Work ............................................................................... 14  
  1.7.1 Quaternionic Gabor Filter for Color Images .................................................... 14  
  1.7.2 Improvement in Face Recognition Accuracy using Color .......................... 15  
  1.7.3 Statistical Selection of Key Face Points ......................................................... 15  
1.8 Organization of this Dissertation ........................................................................ 16  

Chapter 2  
Previous Work  
2.1 Introduction ........................................................................................................... 17  
2.2 Gabor Filters ......................................................................................................... 17  
  2.2.1 Dennis Gabor ................................................................................................... 18  
  2.2.2 Two-dimensional Gabor filters for Images ..................................................... 21  
  2.2.3 Other studies of the two-dimensional Gabor filter ....................................... 24  
  2.2.4 Coherence of the various formulations ......................................................... 24  
2.3 Elastic Graph Matching for Face Recognition .................................................. 29  
  2.3.1 Other Uses of Gabor filters for Face Recognition ....................................... 31  
2.4 Color Face Recognition ......................................................................................... 32  
  2.4.1 Why Color Features are Rarely Used for Face Recognition ....................... 33  
  2.4.2 Eigenface Extension ....................................................................................... 34  
  2.4.3 Other methods ............................................................................................... 34  
2.5 Quaternions ......................................................................................................... 36  
  2.5.1 Background ................................................................................................... 36  
  2.5.2 Basics of Quaternion Algebra ........................................................................ 37  
  2.5.3 Euler’s Formula .............................................................................................. 38  
  2.5.4 Quaternionic Gabor Filter ............................................................................ 39  
  2.5.5 Quaternionic Image Correlation and Convolution ....................................... 39  
2.6 Conclusions .......................................................................................................... 40  

Chapter 3  
Quaternionic Image Encoding  
3.1 Introduction .......................................................................................................... 41  
3.2 Pixel Representations ............................................................................................ 42  
  3.2.1 Color Pixels as Pure Quaternions ................................................................... 43  
3.3 Quaternionic Fourier Transforms ....................................................................... 43  
3.4 Convolutions using quaternionic color pixels .................................................... 44  
3.5 Conclusions .......................................................................................................... 46  

Chapter 4  
Extension of Gabor Filter to Quaternions  
4.1 Introduction .......................................................................................................... 47  
4.2 Four Possible Models ............................................................................................ 47
4.3 Direct Substitution of $\mu$ for $j$ ........................................................................................................... 47
4.4 Extension into Two Complex Axes ........................................................................................................... 48
4.5 Geometrical Interpretation of the Product ................................................................................................... 49
4.6 Opponent Color Feature Model .................................................................................................................. 50
4.7 Properties as a Filter ................................................................................................................................. 51
4.8 Properties as a Wavelet .............................................................................................................................. 52
4.9 Conclusions .................................................................................................................................................. 58
Chapter 5 Optimal Color Projection for Color Images .................................................................................. 60
5.1 Introduction ................................................................................................................................................ 60
5.2 Background and Related Work .................................................................................................................. 60
5.3 K-L Analysis ................................................................................................................................................ 62
5.3.1 Analysis of RGB Color Images ............................................................................................................. 65
5.3.2 Analysis of L-a-b Color Images ............................................................................................................. 67
5.4 Line-fit Analysis .......................................................................................................................................... 69
5.5 Eigenface Performance comparison ........................................................................................................... 69
5.6 Conclusions ................................................................................................................................................ 73
Chapter 6 Quaternion Elastic Graph Matching ............................................................................................... 74
6.1 Introduction ................................................................................................................................................ 74
6.2 Selection of Lattice points .......................................................................................................................... 75
6.2.1 Theoretical Justification ......................................................................................................................... 77
6.2.2 Implementation of the Analysis ............................................................................................................. 80
6.3 Constructing a Model .................................................................................................................................. 82
6.4 Enrolling a New Face .................................................................................................................................. 85
6.5 Matching a New Face .................................................................................................................................. 86
6.5.1 Verification and Identification matching ............................................................................................. 86
6.5.2 Distance measure used in matching ....................................................................................................... 87
6.6 Conclusions ................................................................................................................................................ 88
Chapter 7 Testing Methodology and Implementation ...................................................................................... 90
7.1 Introduction ................................................................................................................................................ 90
7.2 Testing Strategy ......................................................................................................................................... 90
7.3 Software Implementation ........................................................................................................................... 91
7.4 Face Databases .......................................................................................................................................... 92
7.4.1 The Ljubljana CVL database .................................................................................................................. 93
7.4.2 The University of Essex database ......................................................................................................... 93
7.4.3 The CMU PIE database ....................................................................................................................... 94
7.5 Face Image Registration ............................................................................................................................ 94
7.6 Conclusions ................................................................................................................................................ 95
Chapter 8 Performance Comparison .............................................................................................................. 97
8.1 Introduction ................................................................................................................................................ 97
8.2 Comparison of Verification Performance .................................................................................................. 97
8.3 Comparison of Identification Performance ............................................................................................... 103
8.4 Conclusions ................................................................................................................................................ 105
Chapter 9 Conclusions ..................................................................................................................................... 106
9.1 Summary .................................................................................................................................................. 106
9.2 Thesis Statements ..................................................................................................................................... 106
9.2.1 Use of Color Information Can Increase Accuracy ............................................................................... 107
9.2.2 Extending Gabor Techniques to Color Can Increase Accuracy ......................................................... 107
9.2.3 Statistical Methods Can Improve Face Landmark Location .............................................................. 108
9.3 Future Work ............................................................................................................................................. 109
Appendix A - Notable Face Recognition Algorithms .................................................................................... 112
A.1 Eigenface .................................................................................................................................................. 112
A.1.1 Basic Method ....................................................................................................................................... 112
A.1.2 Adaptations and Variations .................................................................................................................... 113
A.1.3 Advantages / Disadvantages ................................................................................................................ 113
A.1.4 Commercial Use .................................................................................................................................. 114
List of Figures

Figure 1 - The Steps of Pattern Recognition................................................................. 3
Figure 2 - Block Diagram of a Typical Face Recognition System.................................. 5
Figure 3 - Intersection of Related Work........................................................................ 17
Figure 4 - One-dimensional Gabor filter with scale = 16............................................. 19
Figure 5 - One-dimensional Gabor filter with scale = 8\sqrt{2}.................................... 20
Figure 6 - One-dimensional Gabor filter with scale = 8.............................................. 20
Figure 7 - One-dimensional Gabor filter with scale = 4\sqrt{2}.................................... 21
Figure 8 - One-dimensional Gabor filter with scale = 4............................................. 21
Figure 9 - Plot of typical 2D Gabor filter and its Fourier transform............................. 22
Figure 10 - monochrome and color face images of the same subject............................ 35
Figure 11 - Three types of receptive fields in the human retina (duplicated from [Jain97]) 51
Figure 12 - Illustration of image extent to be processed for color conversion and recognition 64
Figure 13 - Principal component directions in spherical coordinates for several histograms in RGB space 66
Figure 14 - Principal component directions in spherical coordinates for several histograms in L-a-b space 68
Figure 15 - Comparison of cumulative eigenvalues for the eigenface procedure............. 71
Figure 16 - Result of Lattice Selection Statistical Analysis (Essex database).................... 81
Figure 17 - Sample Image with Face Model Graph...................................................... 85
Figure 18 - Canonical Face Image as described in [Griff02].......................................... 94
Figure 19 - Image Rotation, Scaling and Cropping (from [Griff02]).............................. 95
Figure 20 - ROC curve for Monochrome Verification................................................ 100
Figure 21 - ROC comparison - color vs. monochrome.............................................. 101
Figure 22 – Measured ROC compared with monochrome FRVT2002 results from [Phill03] 102
Figure 23 – the first eight of a typical set of eigenfaces............................................. 113
Figure 24 – Selection of Local Features...................................................................... 115
Figure 25 - Neural network with two hidden layers.................................................. 116
Figure 26 - UML Classes for Basic Types.................................................................... 127
Figure 27 - UML Classes for Jets.............................................................................. 128
Figure 28 - UML Classes for Gabor Filters............................................................... 129
Figure 29 - UML Classes for Face Model, Face Database and Candidate List............... 130
List of Tables

Table 1 - Published formulations of the two-dimensional Gabor filter ................................................................. 26
Table 2 - Harmonization of formulations of the Gabor filter ................................................................................ 27
Table 3 - Eigenvalues for a typical face database using K-L to determine the RGB conversion .............................. 70
Table 4 - Eigenvalues for a typical face database using the Line Fit method to determine RGB conversion ............. 71
Table 5 - Improvement in face recognition performance with new color conversion procedure ......................... 73
Table 6 - Summary of face databases used in this research ................................................................................... 93
Table 7 - EER comparison - color vs. monochrome ................................................................................................. 102
Table 8 - Comparison of Identification Accuracies ............................................................................................... 103
Notation

The following notational conventions and symbols are adopted throughout this document for clarity:

- real and complex variables are shown in italics, either lowercase or uppercase (usually for constants): \( a, M \)
- vectors are shown with an arrow below: \( \vec{v} \)
- matrices are shown in underlined italic capitals: \( \underline{M} \)
- quaternions are shown with a line beneath their name: \( q \)
- \( j \) is used for the single root of \(-1\)
- \( i, j \) and \( k \) are used for the three orthogonal roots of \(-1\), used in quaternions
- \( \in \) - “element of”
- \( \notin \) - “not an element of”
- \( \ni \) - “such that”
- \( \forall \) - “for all”
- \( \therefore \) - “therefore”
- \( \exists \) - “there exists”
- \( \subset \) - “a proper subset of”
- \( \subseteq \) - “a subset of”
- \( \not\subseteq \) - “not a subset of”
- \( \mathbb{R}^n \) - the \( n \)-dimensional space of real numbers
- \( \mathbb{C} \) - the set of all complex numbers
- \( \mathbb{O} \) - the set of all quaternions
- \( \mathbb{O} \) - the set of all octonions
Chapter 1  Background & Motivation

1.1 Introduction

Face recognition is the task of acquiring images of human faces using a camera and matching data from the images to data collected previously, for the purpose of confirming or determining the identity of an individual. Face recognition is a highly visible and important instance of applied pattern recognition. However, the limited accuracy of the technology has prevented its use in many important situations. Several different techniques, based on significantly different approaches to the underlying problem, are used today. All common situations use only monochrome image information, for a variety of non-technical reasons. However, there is the potential for significant improvement in accuracy from the use of color face images.

This dissertation presents a novel extension of a well-known method of feature extraction for face recognition using two-dimensional Gabor filters. This extension makes use of hypercomplex numbers called quaternions, to allow application of the novel filter to color images. This first chapter describes the face recognition application including its current shortcomings, the three thesis statements that are investigated in this research, and the major contributions of this dissertation.

1.2 The Need for Face Recognition

Automated recognition of humans using images of the face is a potentially powerful application of image processing and pattern recognition. For over two decades, significant research has been conducted into methods and algorithms for accurate verification of identities from face images. Most of this effort has been motivated by the desire to apply face recognition to access control, electronic commerce and man-machine interface applications. The technology field of biometrics has grown from this work (along with related technologies for identifying humans using information extracted from fingerprints, iris images, speech patterns, hand geometry and other physiological or behavioral traits).

However, on September 11, 2001, the world changed. The terrorist attacks on the United States gave rise to an immediate and strong call for greater security of the homeland,
exemplified by the creation of a new cabinet-level Department of Homeland Security within the Executive branch of the United States government. It is now widely agreed that technologies for accurately identifying individuals can have tremendous positive impact on the safety of transportation systems, critical infrastructure and other facilities and law enforcement personnel and processes.

Concerns have been voiced about the proliferation of new technologies for human identification. Obviously, appropriate safeguards are vital to avoid negative societal impacts from more advanced identification technologies. These safeguards are a matter of legislative, regulatory and industry standards efforts. Issues related to the appropriate use of face recognition technology (and any automated identification technology) are very important, but will not be considered in this dissertation. This investigation into the how and what of face recognition algorithms has not been carried out in ignorance of the ethical considerations of such technology.

In fact, the author believes strongly that greater accuracy in the face recognition process will actually increase the social acceptability of the technology. The task of automated face recognition is difficult and current systems are not perfect. For the matching of a candidate face against a population of significant size, the best of today's systems have accuracies of perhaps 90% at best. This has serious impact on the usefulness, and therefore the acceptability, of such systems. Consider an automated face recognition system used at an airport gate to identify potential terrorists from a database of face images; the task is to compare each passenger’s face with the faces in the database. There are two types of errors in a biometric system of this type (being used for identification): a false match (sometimes called false positive), in which an innocent passenger is incorrectly identified as a wanted terrorist, and a false non-match (sometimes called false negative), where a wanted terrorist is not detected and is allowed to board the airplane. A false match is a great inconvenience; the innocent passenger is delayed, searched, questioned and may miss the flight, but a false non-match is much more serious. High-security applications typically are configured to reduce the probability of a false non-match to a very small figure, but this inevitably results in a greater number of false matches. The possibility exists for system managers to reduce sensitivity to avoid complaints from travelers, thereby creating security vulnerability. If we are able to
increase the overall system accuracy, we can then decrease the number of false matches without decreasing the overall security; thus, a more accurate technology is seen by the public as less offensive, and has fewer undesirable social impacts. This is not unlike the old proverb that a dull knife is more dangerous to use than a sharp one.

With this in mind, consider the current state of the art in automated face recognition.

### 1.3 Current Face Recognition Technology

#### 1.3.1 Background

This section briefly discusses current technology for face recognition as presented in the academic literature and implemented in the industry. It is difficult to know specifics about how algorithms are implemented in these commercial products, how parameter optimization has been done, and what shortcuts are taken to reduce processing time. More complete analyses of face recognition methods can be found in [Zhao00] and [Gong00].

Note that face recognition is one of the classic pattern recognition problems. It can be thought of as a sequence of the following operations:

![Figure 1 - The Steps of Pattern Recognition](image)

Preprocessing is sometimes as simple as spatial filtering to reduce noise and dependence on precise registration. Classification is usually one of a number of standard methods; common examples are minimum-distance classifiers, Artificial Neural Networks, etc. Feature extraction is the area that tends to differentiate the various methods.

To perform face recognition on real-world data, three major sources of corruption need to be addressed: pose, illumination and expression. Pose is the term for the variations in the position and tilt of the face. Usually, the orientation of the face must be constrained to lie within a small angular range of the normal view, and that the system perform an “affine” transformation on the captured image to correct the scale and rotation of the face image, to
match images in the stored database. Illumination variations tend to be large, due to the shape
of the face (shadows from the nose are a common problem) and occur in both intensity and
color. Both the type and the placement of the illumination source(s) can have a strong effect
on the image. Expression variations are difficult, since they cause a primarily non-systematic
effect on different parts of the face, and are different from person to person.

Another complication that can impact face recognition systems, and that requires high levels
of tolerance to variation, is that systems are often used with non-cooperative or unknowing
subjects. The major advantage to the use of face recognition is that it is the most non-invasive
of all biometric modalities. Cameras can be mounted in such a way that subjects are unaware
that they are being imaged; this is critical for use in security scenarios. However, the
algorithms used are required to tolerate large amounts of variations in head angle and
illumination. In other applications, such as a law enforcement booking or "mugshot" system,
the subject may be non-cooperative and may do anything possible to avoid a match with the
stored database. This makes insensitivity to head angle and expression vital.

This dissertation concentrates on algorithms for recognition of faces in frontal face images.
Any complete face recognition system must also perform face finding, for which color is
frequently used (to locate areas of “skin color” in the candidate image). Face finding is a very
active area of research, and is not addressed in this document. More complex sensing
arrangements offer a richer feature set for face recognition: near-infrared (fused with visible
imagery) and 3-dimensional imaging are two that have great potential and are subjects of
active research. I will not consider them here, since for the near future the cost of sensing
equipment is prohibitive.

Figure 2 shows the block diagram of a typical face recognition system, indicating the aspects
of the system that are investigated in the research work described by this dissertation.
A face image of a subject is acquired using a camera. The image is registered and normalized spatially to best conform to the images in the database. A feature-based representation is computed from the registered subject image, and the features are compared to a number of feature templates in the database. These templates contain the features derived from earlier images of one or more individuals. The best match is chosen and the degree of match is compared to a threshold. If the match is close enough, the subject image is identified as belonging to the individual whose template produced the best match. In identification processing, the subject image is compared to all templates in the database to determine which, if any, individual in the database corresponds; this is sometimes called 1-to-many matching. Verification processing is the case when the subject's identity is claimed or otherwise postulated, and the system only compares the subject image to a single template to confirm or refute the claimed identity; this is also called 1-to-1 matching.

In order for face recognition to take place, a previous image of the subject must have been captured and some information stored to allow subsequent recognition of the same individual. This process of capturing an original image and storing relevant information is called
enrollment. Typically, the information stored is not the original (or raw) image, but a feature-based representation of the image that is sometimes called a model. All commercial systems use models whose specific format and contents are proprietary; even those whose technology is based on well-known methods. A reliable and compact feature representation for face recognition is one of the key components of intellectual property in the face recognition industry.

Appendix A describes in detail the four major areas of current and recent work: eigenface, local feature analysis, neural networks and Gabor wavelet methods.

1.3.2 FERET/FRVT Tests

Beginning in 1994, US government efforts were initiated to test face recognition technology. Initial efforts took place within the Department of Defense; today, the National Institute of Science and Technology (NIST) is tasked with evaluating various technologies for application in support of security legislation, most notably the US Patriot act. As part of this charter, NIST has conducted or supported several rounds of tests of face recognition in the past ten years. Some of the overall results of this testing is presented below, to provide an understanding of the current capabilities of face recognition in actual application.

The first large-scale test of face recognition technology was conducted in 1994-1996 by the DOD Counter-Drug Technology Development Program Office, and was called the FERET (FacE REcognition Technology) test. The FERET tests are notable not only because they established a base understanding of the performance of face recognition, but also because of the well-designed test methods and protocols [Phil96], which are still used in designing biometrics tests today.

The results of the initial FERET testing were not widely seen as encouraging [Phil96]. Many of the reported results were concerned with the impact on accuracy of overall illumination variations (reducing illumination by as much as 60%) and scale variation (up to 30%). The results for each match attempt were reported in terms of a gallery; the matching returned a set of candidates, ranked by score. If the correct match was within the top $n$, then the match was considered a success for gallery sizes less than or equal to $n$. Overall matching results for the algorithms tested were approximately as follows: for one-to-many matching with a total
database size of 831 subjects, correct identification as the top candidate (gallery size = 1) varied from nearly 80% for the best algorithms to less than 40% (!) for the poorest case. One-to-one matching scores ranged between 90% and 70%. When we recall that real-world systems generally require databases much larger than 831 subjects, and that 80% correct identification implies either 20% of the benign population is misidentified as a match, or that there is a 0.2 probability that a person of interest will not be detected, it is seen that these results do not indicate that the technology was ready for real-world use in 1996.

Under the auspices of NIST, and building on the methods of the FERET tests, FRVT (Facial Recognition Vendor Test) were conducted in 2000 and 2002. One notable difference is reflected by the word “Vendor”. In 1994 and 1996, the algorithms evaluated were primarily supplied by academic researchers at the Massachusetts Institute of Technology, the University of Southern California, Rutgers University and other institutions. By 2000, there were several commercial entities with face recognition technologies, and the five participants were all from industry (most were founded, however, on technologies developed at the institutions that participated in the FERET tests!).

Results of the FRVT 2000 reveal some degree of improvement in the capability of the face recognition technologies tested [Black01]. On a gallery of 722 images, the technologies tested reliably achieved between 56% and 63% match accuracy in a one-to-many situation, on the same images used in the FERET testing. However, this is still not indicative of a highly accurate technology.

The FRVT2002 testing [Phil03] had ten participating vendors, only two of which had participated in the FRVT testing two years before (showing the volatile nature of this commercial area). The 2002 testing extended the earlier efforts in several ways: systems were tested on larger databases, up to 37,000 images; variations in lighting and pose were introduced by acquiring images at different times and locations, rather than in a more artificial manner; images extracted from video streams were used for some of the tests; and more automated procedures using XML schema were adopted. Results of this round of tests, broadly summarized, were that correct match percentages ranged from 84% down to 20% for a database of size 1000, but accuracies dropped to a range of 73% to 18% for the full database.
of 37000 images. Significantly, the systems tested grouped neatly into the “top three” whose accuracy only differed by a few percentage points for nearly all tests, and the “lower seven” whose top performer was usually around 10% less accurate than the least of the top three. It so happens that each of the top three vendors use a different basic algorithm as described above: Identix uses a form of Local Feature Analysis, Cognitec uses (it is believed) a form of graph matching and Eyematic uses a Gabor Wavelet transformation followed by a Neural Network for classification.

It is readily seen that face recognition technology has experienced a moderate increase in accuracy over the past ten years, but the current best commercially available technologies are still limited in three areas:

1. **Match accuracy.** The ability to measure the degree of match of two face images or face feature sets with high reliability is still less than desired. Additionally, the current rate of accuracy at identifying a face image against a set of known images (or, just as important, determining that there is no match) is less than 90% under all but the most ideal conditions (small databases, very similar imaging situations, no pose or illumination variation, etc.).

2. **Sensitivity to sources of error.** The current methods’ accuracies degrade quickly when there are variations in pose, illumination, expression, garments and aging of the subject.

3. **Database size.** The most powerful applications of automated identification would make use of large collections of subjects; one example is the currently planned use of face recognition to secure passports, where the database of passport holders would easily be in the **tens of millions**. Note that greater accuracy would allow for use with larger databases, for the same resulting error rates.

Extensive research efforts are underway to find ways to improve the accuracy of the most popular face recognition methods. However, what is needed is not an incremental improvement, but a substantially new method of significantly reducing errors.
1.4 Limitations of Computational Face Recognition Systems

Automated face recognition is not yet able to satisfy the requirements of many important applications, because of limitations on its accuracy. Many of these limitations are associated with specific complications of the real-world use of such systems.

1.4.1 Peak accuracy under ideal conditions

Research publications often report recognition accuracy from a test involving a number of constraints – in image content, variability and population. Many studies of new algorithms report identification percentile accuracies on small-sized populations in the mid- to upper 80's, and verification percentile accuracies in the mid-90's. The needs of conducting an academic performance test make it necessary to establish some constraints on the images used, and therefore will not, in general, produce performance numbers similar to that achievable in real-world conditions. Even without relaxing these constraints, nearly all levels of performance reported in the academic literature are not sufficient for some applications (for example, security check at airport gates).

Through a combination of techniques and population-based optimizations, the best commercial systems can achieve identification rates in the mid-90's and verification rates in the high 90's.

1.4.2 Pose - illumination – expression

The three most well-known external effects on face recognition performance are *pose* (head angle and position), termed *pose*, illumination and expression. In fact, they are so often discussed together that they are sometimes referred to as P-I-E. Each has been shown to reduce matching accuracy when there is variation between the enrollment and matching images.

Pose includes variation in face size, location and angle in all three directions (roll, pitch and yaw). Size can have a strong influence on the function of the matching algorithm; usually, the face image is normalized to a standard size; the distance between the eye centers is often the metric for normalization. Face rotation in the image plane is also normalized so that the line between the eye centers is horizontal. Rotations in the other two axes cannot be entirely
eliminated by processing of the image. Many standards and best practices [INCIT04] for face recognition limit these rotations to ±15 degrees.

Illumination is the most widely analyzed variation. Some of the earliest face recognition algorithm research, based on the eigenface algorithm, revealed a strong sensitivity to illumination variations, since the recognition features were a linear sum of some of the original gray-scale pixel values. One of the justifications for use of Gabor filter responses as face representation features was a high degree of insensitivity to illumination changes ([Gong00]).

Expression, the third variational factor, is less well analyzed. Without a standard method of describing facial expressions, it is difficult to measure the relationship between expression and matching performance. Some biological studies ([Li94], [Daugm80]) have suggested that Gabor filters are less sensitive to influence from variations in expression, probably because they are responsive to changes at many different spatial scales.

1.4.3 Aging

The FRVT 2002 test ([Phill03]) demonstrated that significant amounts of time between initial enrollment and subsequent matching have an impact on the accuracy. Comparison of two face images or feature representations that are separated in time is not as accurate, since many of the features are subject to deformation from aging. In the FRVT 2002 test, data showed that the decrease in accuracy was approximately 5% per year.

While the research described in this dissertation did not address aging, a richer feature set (obtained by including color attributes of the image) may contain elements that are less prone to aging. Intuitively, it seems that the face features that change most with age are intensity-based (wrinkles, jaw profiles, shape of the cheeks and eyes), and that color information may be more stable.

1.4.4 Other Potential Sources of Variability

There are other complicating factors associated with face images taken in real use situations. Glasses obscure some or most of the eye area which is important for both face registration and recognition. Worse, subjects will sometimes have glasses on or off at different imaging times. Likewise, cosmetics, clothing and headwear can vary significantly between images of the
same person taken at different times. These factors are especially important because they impact system accuracy in a way that can be influenced by the subject. If a non-compliant subject desires to lower the probability of a match to the database, they can do so by wearing glasses, different cosmetics and a hat. This can be prevented by choosing face features that are less prone to obscuration and modification (generally those that are closer to the center of the face), and by restrictions on the use of glasses and headwear.

1.5 Current use of color

One potential area of improvement relates to the amount of information contained in the face image itself. Surprisingly, an extensive literature review has discovered no commercial face recognition systems that use color image data for recognition, and only one report of research findings on the application of color features to the actual recognition of faces [Torre99]. In a few instances, color information is used for some preprocessing tasks, as described below. Experts in the field believe that color image information has little value to offer in compensation for the added complexity and processing burden. Some very informal tests have indicated that accuracy improvements from use of color would be small at best [Albio01]. The most compelling reasons to avoid color, from a commercial point of view, are that any system developed would be incompatible with large existing databases of monochrome images [Griff03], and that existing software systems would need to be rewritten to take advantage of color data.

1.5.1 Face Detection

By recognizing regions of skin color in images, it is possible to detect face candidates in images or video streams. This approach is popular in the academic literature [Hsu02], but less so in commercial systems. Most published methods perform a (often nonlinear) color-space conversion to bring all common skin hues into adjacency within the new color space. Then, by segmenting color images using a boundary defined by this region of skin hues, candidate faces are identified. Subsequent processing qualifies each candidate face by aspect ratio, presence of eyes and other face features, and contextual clues within the image. These methods of face detection show great promise, especially since the major sources of error in commercial face recognition systems are face detection and eye location. Note that even
when color is used to detect regions of skin and therefore face candidates, only monochrome data is used for subsequent processing and recognition.

### 1.5.2 Eye Location

Consider that the face appears in an image as a large region of skin hue, with the eyes appearing as two areas of white (low color saturation, high intensity) with darker centers that are roughly aligned vertically. Use of color for eye location is less common than the use of color for face detection; the white portion of the eyes has as much likelihood of detection in monochrome images as in color.

However, there are some definite clues to eye location from the color information in a face image. Hsu et al ([Hsu02](#)) describe the use of color images in Y-C<sub>b</sub>-C<sub>r</sub> format for eye detection. This color format represents color values using one component for brightness and two components to specify the color in a plane of hues. The area surrounding the eyes has been observed to have high C<sub>r</sub> and low C<sub>b</sub> values. To obtain reliable and consistent eye candidates, morphological operations are used for elimination of artifacts and spurious gaps. Lee ([Lee96](#)) and Shioyama ([Shioy00](#)) also describe the use of color to locate eyes and other facial features; Lee concludes that the HSI color representation (with one component each for hue or color, intensity, and saturation or "vividness" of the color) is suited for feature location, while Shioyama supplements a Gabor feature vector with the projection of the color vector onto the H-S plane for characterizing face feature candidates.

### 1.5.3 Extension of the Eigenface Method

The published literature contains almost no work on the use of color for actual recognition of faces. The only significant study had to do with the extension of the eigenface method to color images. Torres et al ([Torre99](#)) made the natural step of extending Principal Component Analysis to color by including all three color planes in the image vectors used to define the starting vector space. Torres uses the YUV color space, in which the color pixel is represented by one component of image intensity and two components to specify the color information. Since the vectors now have three components for each pixel in the input images, the matrices are nine times larger. Nonetheless, once the computationally expensive matrix inversion is done, the recognition process is not significantly more burdensome. Their results show an
improvement of nearly 3.5% - from 84.75% to 88.14% - on an admittedly small sample of 59 faces. This technique suggested that use of color had the potential to increase recognition accuracy, and that other techniques may lend themselves more readily to use with color images.

1.6 Thesis Statements

The surprising absence of face recognition algorithms that make use of color information was the impetus for the research that is described in this dissertation. After some preliminary examinations, the following three thesis statements were developed:

- The use of color information can increase the accuracy of face recognition;
- Extension of Gabor-based feature analysis to color can provide accuracy improvement;
- Statistical analysis, using information-theoretic concepts, can give superior location of key points on the face for feature extraction.

1.6.1 Use of Color Information Can Increase Accuracy

The initial thesis statement that prompted this research is: use of color information can provide increased accuracy in automated face recognition. As described more fully in Section 2.4.1, there are numerous reasons stated for the nearly universal use of only intensity features for face recognition, among which is the belief that there is little advantage to use of color. Examination reveals that this belief is not based on any well-documented research.

Many processing techniques have been inspired by analogy to biological vision. Since human vision makes use of color, at even the lowest levels, it seems appropriate to use color information for tasks that human vision routinely (and accurately) accomplishes, such as recognition of individuals from their face. Yip and Sinha in [Yip02] conclude that color is useful in human face recognition, especially where other image characteristics such as resolution are degraded.
1.6.2 Extending Gabor Techniques to Color Can Increase Accuracy

Based on existing work in monochrome applications of Gabor operations for face recognition, it is suggested that extending the Gabor techniques to color images will provide good performance in analyzing face images. Gabor techniques are not only founded on current understanding of the vision process in mammals ([Jones87]) but they are among the most accurate commercial systems available today. To define a Gabor filter for color images, I will use the concept of the quaternion, a four-component hypercomplex number first defined by Sir William Hamilton ([Hami44]); (refer to Appendix B - Quaternions).

1.6.3 Statistical Methods Can Improve Face Landmark Location

Graph-based methods of face recognition extract features at a number of pre-determined locations on the face. The locations to be considered, called "face landmarks", are generally placed manually and in an ad-hoc manner. Since the extraction of features is a data reduction, it is clear that the features chosen to represent the face image should be the most distinctive (in some analytic sense). It is proposed that a statistical analysis of the entire image area of a complete set of images will provide a near-optimal set of face landmarks.

1.7 Contributions of This Work

The following are the major contributions of this research work.

1.7.1 Quaternionic Gabor Filter for Color Images

Gabor analysis typically consists of the application of a complex filter or transform to a single-valued signal, producing a complex result. The magnitude and phase components of the result are interpreted as describing the similarity of the original signal to the filter applied, or further processed to compute expansion coefficients in the basis space of the vectors defined by the filter(s) used.

For color signals, there is no clear path to apply Gabor filters (or, in fact, any usual complex functions). A significant contribution of this work is an extension of the Gabor filter definition to a four dimensional space, where it is used to operate on a three-dimensional color image. The mathematical operations used obey the rules of hypercomplex numbers; that is, the operations are over the Clifford Algebra given by the space of quaternions $\mathcal{Q}$. Several
different formulations are possible; four are explored here and one is chosen for use in the development and testing.

1.7.2 Improvement in Face Recognition Accuracy using Color

The original motivation of this research was the belief that the use of additional information available in color face images could facilitate higher face recognition accuracy than monochrome information alone. By measuring performance of a parallel implementation of a face recognition testbed, designed to operate with either monochrome images (using complex Gabor filters) or with color images (using Quaternionic Gabor filters), it is shown that measurable (and in some cases, significant) accuracy increases are achieved, for both 1-to-1 and 1-to-many matching.

Since most stated reasons for resistance to the use of color for recognition in the commercial arena are non-technical (or at least non-algorithmic), it may be that adoption of color recognition is still an open question. However, the results of this research provide an indication that performance improvements are available by the use of color features.

1.7.3 Statistical Selection of Key Face Points

Gabor-based face recognition methods are based on extraction of image spatial frequency information at certain key points on the face. Only certain points are used for two reasons: efficiency, and the need to produce a feature set with high discrimination between different individuals, even for varying pose and illumination. In current face recognition systems, these points are determined manually by heuristic methods. This report describes a method for selecting areas on the face that provide high discrimination as determined by the properties of the image pixel values across an image set. Briefly, these are identified as image areas that have high intensity variance for images of different subjects, and also have low mutual information between the distribution of pixel values for multiple images of the same subject, and the distribution of pixel values for images of different subjects. This approach confirms many of the key points commonly used for Gabor-based methods, and also suggests several other points for use in the analysis.
1.8 Organization of this Dissertation

This dissertation is organized as follows. Chapter 1 has provided an overall background on current face recognition technology and on the motivation for investigation into the use of color features. Chapter 2 describes previous published work in four areas that are fundamental to this dissertation: Gabor filters, the elastic graph matching method for face recognition, use of color for face recognition, and the hypercomplex numbers called quaternions. Chapter 3 introduces the encoding of color images using quaternions. Chapter 4 describes the extension of the complex Gabor filter to quaternions, for use on color images. Four different alternatives are described and explored, and the most promising approach is chosen. Chapter 5 explores the identification of an optimal axis for projection of color images; this axis is used as a key element of the novel quaternionic Gabor filter. Chapter 6 describes a method of face recognition using the quaternionic Gabor filter, based on the elastic graph matching method. It also describes a novel method of locating face features using statistical properties of a collection of face images. Chapter 7 discusses the software implementation used to evaluate the method, and the testing methodology that was used. Chapter 8 presents the results of the performance testing and draws some conclusions from them. Chapter 9 contains the conclusions of the dissertation and re-examines the thesis statements in light of the research and the testing results.

Four appendices contain additional information. Appendix A describes the four most common face recognition algorithms in use today. Appendix B discusses the properties of quaternions. Appendix C contains a demonstration of the Euler identity for quaternions, which is vital to the extension of the Gabor filter. Appendix D is a set of Unified Modeling Language (UML) class diagrams for the software implementation. A full list of references completes this dissertation.
Chapter 2  Previous Work

2.1 Introduction

There are four primary areas of existing work that form the foundation of this research: Gabor filters for feature extraction, face recognition using elastic graph matching, color feature extraction and quaternions. There are some instances of research by other investigators in the areas of intersection between any two of these, but none in any three of these areas. The use of quaternion-valued Gabor filters for face recognition with color imagery involves all four of these areas of prior work. Each area is discussed in this chapter, as a background to the remainder of the dissertation.

2.2 Gabor Filters

The simplest expression of a Gabor filter is a one-dimensional complex-valued function formed by the product of a complex exponential and a Gaussian envelope, with the frequency of the complex exponential and the $\sigma$ of the Gaussian having a specific relationship. It has been used for the analysis of signals. The Gabor filter has also been extended to images, where it is commonly used to characterize image spatial frequency in a localized sense, to
enhance images with known frequency behavior in specific directions (fingerprints, for example), and has been suggested for image compression.

2.2.1 Dennis Gabor

In 1948, while Dennis Gabor was at the Thomson-Houston Co. Research Laboratory in Rugby, England [IEEE70], he developed and published a Theory of Communication [Gabor46]. This extended paper contains several major insights into the analysis of signals, including the description of a basic signal for simultaneous analysis of the time and frequency behavior of functions. Gabor began with the Heisenberg uncertainty principle (or, "Heisenberg's principle of indeterminacy"), and applied it to the examination of time-varying signals:

\[ \Delta t \Delta f \approx 1 \]

defining a relationship between "the uncertainties inherent in the definitions of the epoch \( t \) and the frequency \( f \) of an oscillation".

Gabor’s observation was as follows. Time-domain analysis (for example, multiplying a signal by a succession of impulses) can precisely determine the temporal location of a signal, but tells us nothing about the spectral characteristics of that signal. On the other hand, frequency-domain methods, generally based on the Fourier transform, tell us the spectral component of a signal but do not describe when in the time domain a signal may have begun or ended. In other words, the impulse function has infinite temporal resolution but no spectral resolution, while a sinusoid has infinite spectral resolution but no temporal resolution. Gabor postulated a signal that had the optimal balance of resolution in both domains; it turns out that this signal will have the minimum spectral – temporal width product. He found that "the signal which occupies the minimum area \( \Delta t \Delta f = \frac{1}{2} \) is the modulation product of a harmonic oscillation of any frequency with a pulse of the form of a probability function" ([Gabor46], pg. 435). The signal that Gabor suggested and termed the elementary signal (from equation 1.27 in [Gabor46] and shown in equation 2-2 below) can be formed by the product of a sinusoid and a Gaussian envelope of the proper width. This is now commonly called a Gabor function or wave.
Unfortunately, extensive exploitation of the Gabor wave was infeasible in 1948, and its use was somewhat limited even in the early days of digital signal processing.

When used for characterization of signals, it is common to apply a family of Gabor filters at different multiples of a fundamental frequency; these multiples are called the scale of the filter. The one dimensional Gabor filter is shown in Figure 4 through Figure 8 for five scale values $\frac{1}{2}$ octave apart. The half-octave spacing is common in Gabor analysis of signals; it is related to the finding that successive Gaussian filters whose scale varies by $\sqrt{2}$ are strictly scale equivariant (see [Crowl02]). For each filter, the left plot in the group shows the real and imaginary parts of the filter, the center plot shows the magnitude and phase of the filter and the right plot shows the magnitude of the Fast Fourier Transform of the function. The bandwidth is seen to decrease as the temporal width increases.

\[
\psi(t) = e^{-\pi^2 (t-t_0)^2} e^{j(2\pi f_0 t + \phi)}
\]

(2-2)

**Figure 4 - One-dimensional Gabor filter with scale = 16**
Figure 5 - One-dimensional Gabor filter with scale = $8\sqrt{2}$

Figure 6 - One-dimensional Gabor filter with scale = 8
2.2.2 Two-dimensional Gabor filters for Images

As more powerful computing equipment became more widely available, and applications of image processing and analysis began to be applied, the Gabor filter was one of the one-dimensional signal processing techniques that was adapted to two-dimensional images. The most logical and useful extension kept the notion of a cosinusoid modulated by a Gaussian envelope, but allowed the direction of oscillation to at any angle in the $xy$ plane. This
produced a filter with local support that could be used to determine the image's oscillatory component, in a particular direction at a particular frequency. A typical two-dimensional Gabor filter is shown along with the magnitude of its Fourier transform in Figure 9.

Unfortunately, the Gabor filter is not separable in the general case. Unlike many other useful filters, it cannot be reduced to a pair of one-dimensional filters (one in the $x$ direction and one in the $y$ direction) that can be convolved with an image in succession to produce the same output. The Gabor filter requires a complete two-dimensional convolution, unless it is performed in the frequency domain.

The Gabor filter has been used for three types of image processing task: analysis of general texture information [Dunn94], enhancement of nearly periodic features such as fingerprint ridges [Jain01] and face recognition.

Daugman was one of the first to publish extensive descriptions of the two-dimensional Gabor filter for feature analysis. In [Daugm88] he begins with a description of a neural network for
computing the coefficients of projection of a given image onto a set of complete but non-orthogonal basis functions. Based on then-recent conclusions about the physiology and method of human (and more generally mammalian) vision, Daugman chooses a set of related two-dimensional Gabor functions for his work in deriving image compression coefficients. Measurements of the responses of cortical neurons to light, as a function of spatial offset, resulted in functions with characteristic shapes, similar to a Gabor function. Moreover, neurons had different scales of responses, suggesting a family of Gabor functions at different scales.

The physiological work on the understanding of biological vision and its relation to the Gabor filter is summarized in [Jones97]. Here, Jones and Palmer analyze experimental data from feline striate cortices. The receptive fields of two sets of 36 simple cells were measured to evaluate the response as a function of displacement and stimulus. This data was compared to two-dimensional Gabor filters of appropriate width and orientation. Their conclusion is that the Gabor filter provides a suitable model for the response of cells in the visual cortex. It should be mentioned that the Gabor filters used in their models were not circular, as are all the filters that I am using in this study. Two scale factors were introduced to allow the Gaussian envelope to have an elliptical cross-section, and the aspect ratio (the ratio of the major axis to minor axis) ranged from 0.23 to 0.92.

Daugman's work is motivated by the important fact that the two-dimensional Gabor functions defined in this manner are not wavelets by their formal definition, for the reason that they are not orthogonal over scale and shift ([Daugm88]). This has the consequence that the coefficients of a Gabor expansion of a particular image cannot be obtained by simply forming the inner product of the image with the members of the set of functions. In fact, the process of finding the Gabor expansion coefficients for a general image is computationally intensive; this is the reason for his use of a neural network to find approximations to the optimal coefficients. Daugman also applied the Gabor filter to the characterization of the radial features of the iris for human identification [Daugm00]. By projecting selected portions of the eye onto one or more Gabor functions, the specific pattern of the iris feature is determined by the complex phase of the response. In all cases, his published work considered only monochrome images,
though in some cases (e.g., iris characterization) certain spectral bands were used to increase the contrast of the features concerned.

### 2.2.3 Other studies of the two-dimensional Gabor filter

Many other researchers have studied the application of the two-dimensional Gabor filter to feature extraction from monochromatic images. Groups in the Institut für Neuroinformatik at Ruhr-Universität Bochum and at the University of Southern California have jointly developed a method of object recognition based on the application of a family of Gabor filters at certain locations. The major application of this technique has been to the recognition of faces, as discussed in section 2.3.

### 2.2.4 Coherence of the various formulations

The two-dimensional Gabor filter has been presented in different forms in recent publications; in fact, the lack of a standard definition leads to difficulties in duplicating reported results. Yet, each author has either drawn his or her formulation from another source, or in some cases, independently derived the two-dimensional Gabor from the simpler one-dimensional case of a cosinusoid whose amplitude is modulated by a Gaussian curve. I took a very detailed approach to examining each of the most popular versions of the formula itself, for two major reasons. First, I wanted to increase my understanding and intuitive grasp of the Gabor filter by working with several algebraic versions of it. Secondly, I have encountered several typographical errors in representations of the Gabor filters in the literature. I wanted to make sure that I was beginning with an accurate formulation of the Gabor filter for monochrome images.

Wiskott et al. [Wisko99] write the basic Gabor filter (or wavelet) in terms of the spatial location vector \( \vec{x} \) as follows (throughout, I replace their index variable \( j \) with \( h \), and the complex root \( i \) with \( j \) to avoid confusion with the use of \( j \) for \( \sqrt{-1} \)).

\[
\phi_h(\vec{x}) = \frac{k_h^2}{\sigma^2} \exp\left(-\frac{k_h^2 x^2}{2\sigma^2}\right) \left[ \exp\left( jk_h \cdot \vec{x} \right) - \exp\left( -\frac{\sigma^2}{2} \right) \right] \tag{2-3}
\]

In this formulation, the spatial vector \( \vec{x} \) represents displacements from the center location of the filter, and the vector \( \vec{k}_h \) points in the direction of the sinusoidal oscillation of the \( \text{h}^{\text{th}} \) filter.
Thus, the dot product $\vec{k}_h \cdot \vec{x}$ is the component of the spatial location in the direction of the oscillation; this complex exponential term, then, is the oscillatory component, while the last term is simply an offset so that the Gabor filter always integrates to zero across the entire plane. The first exponential is of order -2 in terms of spatial location and is the Gaussian envelope. This formulation by Wiskott is the clearest and the most convenient to use as a basis.

Liu and Wechsler [Liu01] express the 2D Gabor filter in a slightly different manner:

$$\phi_{\mu, \nu}(\vec{z}) = \frac{\|\vec{k}_{\mu, \nu}\|^2}{\sigma^2} e^{-\left(\frac{\|\vec{z}\|}{2\sigma^2}\right)} \left[e^{j\vec{k}_{\mu, \nu} \cdot \vec{z}} - e^{-\frac{\sigma^2}{2}}\right]$$

where $\mu = \text{orientation}$, $\nu = \text{scale}$, $\vec{z} = (x, y)$, $\vec{k}_{\mu, \nu} = k_v e^{j\phi_{\mu}}$, $k_v = \frac{k_{\text{max}}}{f^\nu}$, $\phi_{\mu} = \frac{\pi \mu}{8}$, $k_{\text{max}} = \text{maximum frequency}$ and $f = \text{spacing factor between different frequencies}$ in the Gabor filter set. This set of relations allows generation of the entire set of desired Gabor filters by specifying the proper values of maximum frequency and frequency spacing. It is important to note that the $j$ in the exponential in the main equation is not in the same sense as the $j$ in the equation for the wave vector $\vec{k}_{\mu, \nu} = k_v e^{j\phi_{\mu}}$. The main equation (2-4) is a complex exponential and the $j$ is the complex root, so that the resulting function is complex. However, the $j$ in the equation for $\vec{k}_{\mu, \nu}$ signifies that the vector points in a direction in the x-y plane determined by the angle $\phi_{\mu}$. This is not clear in Liu and Wechsler’s paper; I have added my vector notation to the equations above to attempt to clarify this. It should be noted that Liu and Wechsler’s tracing of the values of the Gabor filter back to the wave function and the basic constants is very helpful, and is not present in most other publications.

Fasel [Fasel02] writes the Gabor filter as

$$g(x, y) = \exp\left(-j(2\pi(u_0x + v_0y))\right) K \exp\left(-\pi\left(a^2(x-x_0)^2 + b^2(y-y_0)^2\right)\right)$$

with

$$\begin{align*}
(x-x_0)_r &= (x-x_0)\cos\theta + (y-y_0)\sin\theta \\
(y-y_0)_r &= -(x-x_0)\sin\theta + (y-y_0)\cos\theta
\end{align*}$$

and

This has some important differences: the wave
direction vector is written implicitly as \((u_0, v_0)\), there is no offset term to ensure integration to zero, and the scale factors (including \(\sigma\)) are not present.

Smeraldi [Smera00] writes the Gabor filter in the frequency domain as

\[
\tilde{G}(\tilde{w}|\sigma_x, \sigma_y, w_0) = \exp \left( -\frac{(w_x - w_0)^2}{2\sigma_x^2} \right) \exp \left( -\frac{w_y^2}{2\sigma_y^2} \right)
\]

(2-6)

This is interesting because it allows different Gaussian widths in the two spatial directions \((\sigma_x\) and \(\sigma_y\)). Applications for this form might include situations where there is a priori information about the spatial frequency properties of the image that would suggest known and different amounts of smoothing in the \(x\) and \(y\) axes, or situations where the Gabor filters are used to enhance periodic image features. Smeraldi also defines a modified Gabor filter whose extent is non-symmetric in frequency, using the following rationale. Different Gabor filters whose center frequencies are related exponentially cover increasingly wider frequency ranges, so the frequency domain of each should be biased towards higher frequencies to most effectively cover the spectral plane.

In the following table I have attempted to summarize the formulations for the two-dimensional Gabor filter that I found in the most popular publications, along with my attempts to harmonize them. As stated earlier, Wiskott’s equation was used as the starting point for this work; several of the other formulations were more difficult to analyze or implement, or even incomplete.

<table>
<thead>
<tr>
<th>Citation</th>
<th>2D Gabor filter formulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Gabor46]</td>
<td>(\psi(t) = e^{-\alpha^2(t-t_0)^2} e^{j(2\pi f_0 t + \phi)}) (one-dimensional, included for comparison)</td>
</tr>
<tr>
<td>[Wisko99]</td>
<td>(\varphi_h(\tilde{x}) = \frac{k_h^2}{\sigma^2} \exp \left( -\frac{k_h^2 x^2}{2\sigma^2} \right) \left[ \exp \left( jk_h \cdot \tilde{x} \right) - \exp \left( -\frac{\sigma^2}{2} \right) \right] )</td>
</tr>
</tbody>
</table>
Each formulation has the essential characteristics of a complex sinusoid with the argument being position along a line at the specified angle in the $xy$ plane, with an envelope given by a two-dimensional Gaussian filter. Some formulations allow for the Gaussian filter to have different widths in the $x$ and $y$ direction. This additional variability is not useful for the current application of face recognition.

### Table 2 - Harmonization of formulations of the Gabor filter

<table>
<thead>
<tr>
<th>Citation</th>
<th>Function</th>
<th>Scaling</th>
<th>Gaussian envelope</th>
<th>Complex sinusoid</th>
<th>Bias removal</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Gabor46]</td>
<td>$\psi(t) = \exp(-\alpha(t-t_0)^2)$</td>
<td></td>
<td></td>
<td>$e^{i(2\pi f(t-t_0)+\phi)}$</td>
<td></td>
</tr>
<tr>
<td>[Wisko99]</td>
<td>$\varphi_h(\vec{x}) = \frac{k_h^2}{\sigma^2} \exp\left(\frac{-k_h^2 x^2}{2\sigma^2}\right)$</td>
<td></td>
<td>$\exp\left(jk_h \cdot \vec{x}\right)$</td>
<td>$-\exp\left(-\frac{\sigma^2}{2}\right)$</td>
<td></td>
</tr>
<tr>
<td>[Liu01]</td>
<td>$\varphi_{\mu,\nu}(\vec{z}) = \left|\tilde{k}<em>{\mu,\nu}\right|^2 e^{-\left|\tilde{k}</em>{\mu,\nu}\right|^2 \frac{2\sigma^2}{\sigma^2}} e^{j\tilde{k}_{\mu,\nu} \cdot \vec{z}}$</td>
<td></td>
<td></td>
<td>$e^{j\tilde{k}_{\mu,\nu} \cdot \vec{z}}$</td>
<td>$-e^{\frac{\sigma^2}{2}}$</td>
</tr>
<tr>
<td>Reference</td>
<td>Equation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------</td>
<td>----------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lyons00</td>
<td>( \varphi(k, x) = \frac{k^2}{\sigma^2} \exp \left( -\frac{k^2 x^2}{2\sigma^2} \right) \left( \exp(jk \cdot x) - \exp\left(-\frac{\sigma^2}{2}\right) \right) )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Manju92</td>
<td>( g_{x\lambda}(x, y, \theta) = \exp(-\frac{x^2 + y^2}{\lambda^2}) e^{j\pi x^2} )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gong00</td>
<td>( \varphi(x, y) = k \exp\left(-\frac{r^2 (\frac{k}{\sigma})^2}{2}\right) jk \sin \theta \cos \theta \exp\left(-\frac{\sigma^2}{2}\right) )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Daugm88</td>
<td>( G(x, y) = \exp\left(-\pi \left[(x-x_0)^2 + (y-y_0)^2/\beta^2\right] - \exp(-2\pi j [u_0 (x-x_0) + v_0 (y-y_0)]) \right) )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fasel02</td>
<td>( g(x, y) = K \exp\left(-\pi (a^2 (x-x_0)^2 + b^2 (y-y_0)^2) \right) \exp\left(-j(2\pi (u_0 x + v_0 y)) \right) )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Possible errors or areas of imprecision in the literature include the following. In [Wisko99], it is not stated that \( k_j \) is the magnitude of the wave vector \( \vec{k}_j \), though this can be inferred from the text. In [Liu01], the complex exponential should have as its argument the dot product of \( k_{\mu,\nu} \) and \( z \); the dot is missing and the description of \( k_{\mu,\nu} \) does not make it clear that it is even a vector. Similarly, [Lyons00] has the argument of the complex exponential as \( ik \cdot x \); it must be assumed that the period is a typographical error and the vector dot product was intended. Manjunath et al. in [Manju92] define \( x' \) as the dot product of the spatial displacement and the wave vector, but they do not include the proper scale factor or bias removal. This will result in Gabor responses that are of non-unity gain and have a constant offset. The offset may cause problems in the recognition phase, if normalization is not used at some point in the classification. Gong [Gong00] expresses the complex exponential directly in its complex sinusoidal parts, but gives no detail on the scale factor to use for unity gain. Daugman, in [Daugm88] also includes no scale factor or bias correction.

Based on this harmonization, I have used the two-dimensional Gabor filter as formulated by Wiskott et al in [Wisko99] as the foundation for my extension to quaternions. It is the most complete and accurate of the representations in the literature that I reviewed.
2.3 Elastic Graph Matching for Face Recognition

A recently described method of face recognition using two-dimensional Gabor filters is the elastic graph matching method [Brüne01], [Lades93], [Wisko99], [Wisko03]. The fundamental notion is to represent a face by a set of feature vectors called jets, formed by convolving a set of filters with the face image at certain locations. The collection of jets is matched against a number of stored face models, each corresponding to a particular individual. Any of a number of matching strategies may be used.

The most recently described method is called “elastic bunch graph matching” [Wisko03] and can be described as follows. Face images are represented internally by a set of face models. Face images to be used for matching or for generation of a model must be registered; that is, the image is scaled, rotated and shifted to best correspond to the expected location of some landmarks, usually the eye centers. Face registration is discussed more specifically in section 7.5.

Construction of a set of models begins by manual placement of a number of nodes or fiducial points with respect to a registered face image. Locations are chosen that experience and inspection suggest are areas of distinctive appearance on images of different faces. Approximately 40 nodes are defined on the face. On an initial exemplar image, the nodes are very carefully placed to ensure their placement at the intended point on the face. At each node location, a set of 40 different Gabor filters are applied to the image, at 5 different scales (at ½ octave spacings) and 8 different angles (equally spaced over the range of 0 to $\pi$ radians). The responses of the jets are concatenated into the jets. Thus, the face model for a particular image includes a number of nodes, each of which is represented by a location and a prototype jet, as well as a number of graph edges that connect only certain nodes in the model. The nodes are connected where the distance between the nodes is seen to be a good discriminator between different individuals; for example, in most cases the distance between the nose and mouth is a useful feature for recognition. This completes the process of enrolling a new face image.

To add a second face image to the database of “enrolled” faces, a search is conducted in a small neighborhood of each node location to find the point at which the face feature best matches the one(s) already in the database. This allows for the face that different individuals
have face features in different locations. Once the best match to an existing jet for that node has been found, the spatial offset from the nominal location for that landmark becomes part of the face model.

When only one face model is in the database, the search for the best match to the existing jet at each node is straightforward. If many faces are already in the database, the processing time required to compare all potential node locations in a neighborhood to the corresponding node for each face model can easily become excessive. For this reason, the notion of a bunch graph is introduced. As each new node is located, if its jet is sufficiently close to a jet already in the database, then it is simply recorded as another occurrence of the same jet, and the jet features are (optionally) averaged or otherwise consolidated. This will result in the database containing a set of jets for each node that represent the common types of face features at that location, rather than a unique jet for each face observed. This modified database is called the bunch graph, since each node contains a bunch of jets. Thus, nodes in a new face image must only be compared to the number of jets existing in the bunch graph rather than the number of faces in the database.

Matching of a face against the database occurs much like addition of a new face model. First, the face must be spatially registered using the eye locations or other landmarks. In the vicinity of each node, a search is performed to find the location having the best match to any of the jets in the bunch graph. A match is detected by using the following similarity function (from [Wisko99]. For a graph $G$ derived from image $I$ with nodes $n = 1, ..., N$ and edges $e = 1, ..., E$, and a bunch graph $B$ with models $m = 1, ..., M$, the similarity is:

$$S_B((G', B)) = \frac{1}{N} \sum_n \max_m \left( S_{\phi} (J_n^e, J_m^\beta_e) \right) - \frac{\lambda}{E} \sum_e \frac{\left( \Delta x_e^f - \Delta x_e^B \right)^2}{(\Delta x_e^B)^2}$$

(2-7)

with $J_n$ as the jet at node $n$, $\Delta x_e$ as the edge distances in the graph and $\lambda$ as a constant to adjust the relative effect of the jet matching and the distance matching. The report in [Wisko99] reports that graph deformation was not used in the matching process.

The earlier embodiments of this method used jets extracted by the application of Gabor filters at the vertices of a rectangular grid superimposed on the face rather than the more general
graph, and only used the magnitude of the response function at each face location [Lades93]. Adoption of the graph matching and the use of phase information for node localization provided a claimed increase in accuracy of approximately 8%.

The elastic bunch graph method is the subject of much additional work, both by the original researchers and by other groups. It is also used in at least one popular commercial system. The bunch graph methods are covered by a US patent [Wisko03], and form the basis of the technology of Eyematic Inc.

2.3.1 Other Uses of Gabor filters for Face Recognition

Liu and Wechsler [Liu03] describe a synthesis of two existing techniques for face recognition – Gabor filter analysis and the eigenface approach. They apply a set of Gabor filters to the subject images to extract feature vectors. Then, the resulting features undergo principal component analysis and independent component analysis, in which higher-order statistics of the presumed distributions are considered in selecting the most significant components for representation. Finally, they use a probabilistic reasoning model, which postulates a Bayes linear classifier based on some assumptions about the in-class distributions of the transformed (or projected) feature vectors. They report recognition rates of approximately 76% on the FERET database of monochrome images, which contains approximately 1000 images (the precise number depends on exactly which version of the FERET database was used in the testing).

Without endeavoring to evaluate their method, it should be noted that one of the assumptions that they make is that the features extracted by the Gabor filters can be projected (using PCA) onto a space in which they are highly diagonalizable, even in the presence of noise, imaging artifacts such as pose and illumination variation, and that the resulting models will give good recognition of probe (non-taught) images.

Duc [Duc99] implemented a system using the elastic bunch graph method as described above. However, he added a process in which the individual weights applied to each node in the classification step are computed by a discriminant analysis of the faces in the database. The nodes that are more relevant at discriminating the faces in the database are given more weight
in the final classification step. This enhancement provided a notable decrease in the equal-error rate (from 11.8% to 6.1%).

Lyons et al [Lyons00] used a bunch graph method based on Gabor features for face analysis and discrimination. Their application, however, was not to the identification problem but to the determination of various "face attributes" such as race, gender and expression. In contrast to most other uses of the Gabor feature extraction concept, they composed feature vectors using six equally spaced angles (rather than eight); as in most other studies, they used five different Gabor scales separated by one-half octave ($\sqrt{2}$ in frequency). The system was trained on a number of images in each category: male vs. female, Asian vs. Caucasian, and six different expressions: happy, sad, angry, fearful, surprised and disgusted. They obtained good results; over 90% of the probe or live images processed by the system were classified correctly. While this shows the wide application of the Gabor-derived features, it is interesting that they were useful in distinguishing expressions when other studies showed that they had some degree of expression-invariance.

It should be noted that there is no strong proof that the Gabor are the optimal filters for face recognition or indeed for feature extraction in general. Their wide use seems to be based on the simultaneous minimization of spatial and spectral uncertainty, as described by Gabor himself. A second factor in their adoption is the similarity to our current understanding of early vision processing in mammals (see [Jones87]). Computer vision often makes use of biologically inspired methods, even though it seems (to this researcher) that the attempt to accurately emulate human vision is exceedingly daunting. An additional attractiveness is their intuitive sensitivity to periodic behavior of a specific frequency in a specific direction. The eye almost seems to search for regions in the face image where some Gabor response will be high. However, it is quite possible that there are other filters that are in fact superior for the specific characteristics of face images, especially in color. This is an interesting area for future investigation.

### 2.4 Color Face Recognition

The use of color image data for face recognition is the third major area of existing work on which this research was based. As has been stated, there is a surprisingly small body of
published work in this area, and there are no well-known commercial systems that currently use color image features.

2.4.1 Why Color Features are Rarely Used for Face Recognition

In many identity applications today, color imagery is collected as a matter of course. This has been the case since at least the early 1990’s, when the cost of quality color imagers became competitive with black-and-white. Of course, to a human operator, color images are much more useful for identification, since cues such as hair color and complexion can be used. Assuming that the image resolution and noise performance are sufficient (this is not a trivial assumption in real-world systems!), the use of color images to recognize faces automatically should offer benefits comparable to those for manual identification.

However, color images are rarely used for face recognition. Face images are usually collected in color to facilitate other uses in the application – printing of ID cards or driver license documents, for example. In addition, isolating skin tones in the image is a common technique for face location. In almost all cases, the images are reduced to monochrome prior to feature extraction.

Interviews with commercial face recognition algorithm developers have revealed several reasons for the common use of only monochrome features for the recognition task:

1. An unwillingness to develop technology that is inapplicable to existing monochrome databases;
2. The additional processing time required to deal with three times as much raw data;
3. Use of color features introduces a dependence on the “white balance” of the camera (the relative gain and offset properties of the three color channels);
4. Color features of the face are more variable than monochrome features – for example, ladies’ complexions can often change hue due to different cosmetics;
5. A perception, based on very preliminary tests, that no appreciable benefit can be gained by using color features for recognition.

Factor 1 – inapplicability to legacy monochrome databases – is purely commercial in nature and will decrease in importance as time goes on; therefore I choose to ignore it. Factor 2 –
increased processing time – is based on the assumption that each pixel must be processed, and will also decrease in relevance as time and Moore’s law march on. Factor 3 – a requirement to white balance the images – is an important consideration, but if use of color provides some benefit, a need to preserve white balance will be a small burden to bear. Most color cameras in use today do some automatic white-balancing, but it is generally scene- and history-dependent. There will be differences that the recognition system must automatically compensate for. In addition, it is not at all clear that small variations in white balance will be relevant; this will be explored further as the work progresses.

Factors 4 and 5 address the heart of the question – can the additional information provided by the three colors yield additional benefit in terms of accuracy and robustness? “Accuracy” can be considered as minimization of the combination of False Match and False Non-Match errors, as described in section 8.2, while “robustness” is the preservation of accuracy in the presence of variations in lighting, expression and viewing angle of the face.

2.4.2 Eigenface Extension

The familiar eigenface method, based on application of Principal Components Analysis to pixel values in normalized and registered face images, can be extended to color imagery in a very straightforward manner. Torres et al. [Torre99] developed a method of normalizing and computing the principal components in each color plane separately, then concatenating the resulting projections for classification. Their work shows an increase of approximately 3% to 5% in verification accuracy. While this method retains most of the disadvantages of the eigenface method (strong dependence on pose and illumination, reliance on highly accurate face registration and the need for a computationally expensive matrix inversion at the time of creating the database), they do demonstrate the potential improvement in accuracy that is available from the use of color.

2.4.3 Other automated color face recognition methods

Some research efforts into the use of color for other face-related tasks also touch on the use of these techniques for recognition. Lee et al. [Lee96] uses color information in HSI space to find and quantify the eyes, eyebrows and mouth, and their results have likely application for recognition.
2.4.4 The role of color in face recognition by humans

One justification for the belief that color features are useful for recognizing faces is the observation that it is used by the human visual system for face recognition. Examination of side-by-side images of a face (Figure 10), in monochrome and color, reveal that there are features and areas of distinction that are either not discernible or less discernible in the monochrome image. For example, in the color image the precise shape of the hairline is more apparent, the skin under the jaw is seen to be different, the boundary between the lower lip and the skin is more distinct and the detail within the ear is more defined. It is this ability to better define features that most strongly motivates the use of color, and also leads to its inclusion in the most basic feature extraction process rather than a high-level process such as a syntactic image description (the hair is red, the eyes are brown…).

There are published results from psychovisual studies that suggest the value of color to human vision. Yip and Sinha ([Yip02]) have published the results of a study of recognition performance of human observers when presented with images of well-known celebrities. Two groups were tested, one using monochrome images and one with color images. At high resolutions, the accuracies were very similar; recognition rates were approximately 5% better (86% correct versus 91%) when color was used. However, at
very low image resolutions, the improvement due to color was more dramatic; an 8% accuracy for monochrome images improved to 24% when color was used. Interestingly, when they used false-color images in which the hue plane was transformed to produce images where the colors were distinct but unnatural, the accuracy was nearly as high as with the true color images (see [Yip02]). This led them to postulate that the color information was useful in low-level vision tasks such as segmentation and characterization, and was not solely used in what they term "diagnostic" tasks such as "this must be a red-headed person". Not only is this an interesting suggestion as it relates to understanding of human vision, but it also points toward the use of color information early in the automated vision process. This is the approach that I have taken by defining and extracting color features and using a general recognition methodology.

### 2.5 Quaternions

#### 2.5.1 Background

One disincentive to adoption of color information for object recognition is the difficulty in modeling color image information in the general case. Since the information in the different color planes is obviously correlated in natural images, treating them as independent planes is not easily justified. There is a need for a rigorous mathematical method for manipulating color pixel values, typically in three dimensions.

This is the same problem that was considered by Sir William Hamilton in 1843. In the process of researching mathematical methods that are suited for three-element vectors, he had been unsuccessful in developing a formula of the product of two triples (as he called them) that satisfied intuitively important properties of multiplication: existence of a unique multiplicative inverse, any two vectors should only have a product of zero if at least one is the zero vector, and others. The vectors were expressed as a linear sum of three orthogonal components, one real part and two independent roots of -1, based on Hamilton's previous work on complex numbers. He approached the problem [Koets95] by drafting several possible expressions of the product of two vectors in terms of the individual components and examining the possible redefinitions of the components' products.
On October 16, 1843, Hamilton was walking with his wife, when he had the insight that the properties of multiplication that he was seeking would be achieved by the definition of a fourth component, a third orthogonal root of -1. Upon this realization, he stopped and carved the key equation into the stone of Broome (Brougham, to Hamilton) bridge over the Royal Canal in Dublin [Hamil65]:

\[ i^2 = j^2 = k^2 = ijk = -1 \]  

(2-8)

With the introduction of the third root, \( k \), Hamilton was able to define the quaternion algebra.

The general notion of defining vectors in the space formed by several orthogonal roots of -1 is often called hypercomplex analysis. Since Hamilton's time, other possible hypercomplex numbers have been defined that preserve the commutative property of multiplication. However, in these systems, there are some nonzero hypercomplex numbers that do not have a multiplicative inverse. For the purposes of image feature extraction, this is a more severe limitation.

### 2.5.2 Basics of Quaternion Algebra

Quaternions are elements with four orthogonal components obeying the following relations:

\[
\begin{align*}
    i^2 &= j^2 = k^2 = -1 \\
    ij &= k, \quad jk = i, \quad ki = j \\
    ji &= -k, \quad kj = -i, \quad ik = -j
\end{align*}
\]  

(2-9)

Quaternions add by component-wise addition:

\[
\begin{align*}
    p + q &= p_0 + q_0 + (p_1 + q_1)i + (p_2 + q_2)j + (p_3 + q_3)k
\end{align*}
\]  

(2-10)

The product of two quaternions is expressible as:

\[
\begin{align*}
    pq &= \left(p_0q_0 - p_1q_1 - p_2q_2 - p_3q_3\right) + \left(p_0q_1 + p_1q_0 + p_2q_3 + p_3q_2\right)i + \\
    &\quad \left(p_0q_2 + p_2q_0 + p_3q_1 - p_1q_3\right)j + \left(p_0q_3 + p_3q_0 + p_1q_2 - p_2q_1\right)k
\end{align*}
\]  

(2-11)
As can be seen from equation (2-9), reversing the order of multiplication of the imaginary components inverts the sign of the product of those components. For this reason, quaternion multiplication is not commutative; multiplication of pure quaternions (those with no real part) is anti-commutative, but the inclusion of a real part makes the product non-commutative in any sense.

An important result is obtained for the product of two pure quaternions:

\[
pq = -(p, q_i + p, q_z + p, q_x) + (p, q_x - p, q_z)i + (p, q_z - p, q_x)j + (p, q_i - p, q_z)k \quad (2-12)
\]

Quaternions form a Clifford Algebra [Conwa03]. A Clifford Algebra is a system in which the product of two elements has two separate parts: a scalar part equal to the dot product and a vector part related to the cross product of the two elements. This special formulation of product is known as the Clifford product. Quaternions are not the only well-defined Clifford algebra; \( \mathbb{R} \) and \( \mathbb{C} \) (the sets of reals and complex numbers, respectively) are degenerate cases of Clifford Algebra, and the octonions, formed from seven imaginary roots, also form a Clifford Algebra.

Quaternions are only recently being applied to image analysis – both for monochromatic and color images. However, the use of quaternions is common in other areas involving three-dimensional vectors: orbital mechanics, three-dimensional graphics and modeling of surfaces and shading. More detail on quaternions is given in Appendix B - Quaternions.

### 2.5.3 Euler’s Formula

The generalization of the Euler identity for complex numbers to pure quaternions is

\[ e^{\theta} = \cos(\theta) + \mu \sin(\theta) \].

Therefore, exponentiation of any general quaternion \( q \) can be determined as:

\[
e^q = e^{\theta} e^{\text{Im}(q)} = e^{\theta} e^{\text{Im}(q)} \left( \frac{\text{Im}(q)}{\|\text{Im}(q)\|} \right) = e^{\theta} \left( \cos(\|\text{Im}(q)\|) + \left( \frac{\text{Im}(q)}{\|\text{Im}(q)\|} \right) \sin(\|\text{Im}(q)\|) \right) \quad (2-13)
\]
If $\mu$ is a unit pure quaternion, then $e^{i\theta} = [\cos \theta, \mu \sin \theta]$, where the $[]$ notation indicates the real and complex parts of the quaternion: $[\Re(q), \Im(q)]$.

2.5.4 Quaternionic Gabor Filter

Bülow and Sommer define a quaternionic Gabor filter for use with scalar images [Bulow98]. Their thesis is that scalar images have a local structure (the odd and even symmetries of the image in a neighborhood) that can be detected. They extend the Gabor filter to quaternions by using the two quaternion bases $i$ and $j$ to replace the single complex root $j$ in the complex Gabor filter. As the complex Gabor filter is $\mathbb{R} \to \mathbb{C}$, so Bülow’s quaternionic Gabor filter is $\mathbb{R} \to \mathbb{Q}$. Just as in the complex case, the magnitude of the response is indicative of the correlation of the image with the Gabor filter. There are three phase angles associated with any quaternion (interpreting the imaginary part as a vector in $\mathbb{R}^3$), which give some information about the relationship of the response to the input signal in first-order and second-order terms. Since their work has concentrated entirely on mappings for real images, it is not considered further in this research.

2.5.5 Quaternionic Image Correlation and Convolution

To measure the response of the quaternionic-valued filters to an arbitrary image, it is necessary to define correlation and/or convolution for sets of quaternions. A straightforward extension of the usual definition of convolution is given in section 3.5. Moxey, Sangwine and Ell in [Moxey03] define the correlation of a quaternion-valued function with a color image, defined as an image of purely imaginary quaternions. Their definition is in agreement with that stated in section below, as:

$$
\sum_{q=0}^{M-1} \sum_{p=0}^{N-1} f(q, p) g(q-m, p-n)^* \tag{2-14}
$$

The $*$ notation indicates the complex conjugate of the quaternion, obtained by the negation of all three complex components. Their paper makes the observation that a straightforward implementation of (2-14) has the computational complexity of $O(N^4)$ where $N$ is the image size. For this reason, they develop the required Fourier transforms required to perform the hypercomplex correlation by multiplication in the transform domain. They demonstrate that
the usual equality of convolution in the time or spatial domains and multiplication in the Fourier domain; this is a very significant result. The work being described here did not make any use of the Fourier domain to increase efficiency, although this would be a natural approach for a practical system.

Especially notable from the work of Moxey et al. [Moxey03] is their definition of hypercomplex (or quaternion correlation) that is in agreement with the approach taken in this research.

Pei and Chen (in [Pei01]) also define correlation of quaternion-valued images in the same manner. They use the hypercomplex correlation for object matching in color images, and demonstrate the feasibility of this approach.

2.6 Conclusions

There are significant bodies of research work in each of the four basic areas: use of Gabor filters for feature extraction, graph-based face recognition, color image processing and quaternions. Each area will be borrowed from in the development of the concept of quaternionic Gabor filter for color images, and in its application to face recognition.
Chapter 3  Quaternionic Image Encoding

3.1 Introduction
In order to represent color image pixels in a manner that allows application of an extended Gabor filter, this dissertation proposes the use of quaternions. This chapter describes a representation for a color image as a two-dimensional array of quaternions. After the background for this approach is described, the specific assignment of the red, green and blue planes of the image is related to the components of the quaternion-valued image. This is followed by brief discussions of the notion of a quaternionic Fourier transform and the convolution of a quaternion-valued image with a quaternion-valued filter. Finally, the extraction of quaternionic jets is described and conclusions of this chapter are presented.

3.2 Background
A common approach in signal analysis is to characterize a signal in terms of its response to a complex-valued filter, typically by convolution. For example, the result of applying an input signal to a system with known behavior can be found by convolution (in the time domain) of the input signal with the impulse response of the system. The impulse response is a complex quantity, which can be interpreted as the magnitude and phase shift (delay) of the response to an impulse input. Thus, the complex-valued impulse response is suitable for measuring system response to a single-valued stimulus function. Two-dimensional complex-valued functions (filters) are used to extract features from single-valued, or monochrome, images. The additional degree of freedom provides both magnitude and phase information about the relationship of the image and the filter signal.

A color image is generally represented by three components, with each location (or image pixel) encoded by three independent quantities. Common formats for color images are: red, green and blue, (RGB); hue, saturation and intensity (HSI); intensity and a two-dimensional form of the color information (Y-C_r-C_b); and others are known. A typical complex-valued filter is inappropriate for extraction of features from a color image. If each color plane is analyzed individually, we are not making any use of the interrelationship of the planes; for
natural color images, the three color planes are strongly correlated, and the degrees of correlation contain useful information.

Could the usual complex Gabor filters be applied to the three color planes simultaneously to extract information from color images? Yes, but there are several disadvantages to this approach. The mathematics of the process are not well defined, and would consist of some ad-hoc process that is difficult to justify. The resulting responses would be completely separate components, making the final jets too susceptible to variation in color. A concern in the introduction of color information was the need to not introduce new sources of variation or sensitivities to typical behaviors of color images. In the real world, not all color imagers have a good “white balance”; that is, the illumination, sensing and digitization do not necessarily preserve appropriate gain and offset in the three color channels.

An alternative approach is to represent the color pixels as three-component vectors, and the values of the analysis filters as three-component vectors, in some system within which the usual mathematical operations are well-defined and well-behaved. This is precisely the dilemma that faced Hamilton in the mid-1800’s. After extensive work at trying to formulate sensible algebraic operations for three-component vectors, he made the discovery that inclusion of a fourth orthogonal component would allow a viable algebra (see [Conwa03]). This is the same insight that leads to the use of Hamilton’s solution – the quaternion – as the mode of representation of color images and suitable filters for color images (see section Appendix B - Quaternions).

### 3.3 Color Pixel Representations

Since the quaternion includes four components, and a color pixel in RGB or other space contains three components, it is necessary to choose an assignment of the color planes to the quaternion elements. The three imaginary axes of the quaternion space are orthogonal and there is no reason to make any difference between the three of them, other than possible consideration of the chirality ("handedness") of the coordinate system. The essential question is whether any of the colors should be assigned to the real component of the quaternion. Frequently, quaternions are used to model three-space vectors with the real component assigned to a related but non-orthogonal quantity such as time.
### 3.3.1 Color Pixels as Pure Quaternions

Generally, the different components of a color image are processed separately, and this has led to a tendency to interpret them independently. For real images, however, the content of the different color planes are obviously correlated. The contours of a real object will in general cause intensity variations, indicative of edges, in all three planes. Therefore, color representations that fail to use some sort of joint representation are not optimal.

Note what happens when the three components of the color image pixel are assigned to the three imaginary portions of the quaternion (assuming for sake of notation that the RGB color space is used). From equation 2-11, the product of a quaternion and a three-valued vector representing a color image pixel is:

$$p_{rgb} \cdot q = -\left( p, q_1 + p, q_2 + p, q_3 \right) + \left( p, q_0 + p, q_3 - p, q_2 \right)i$$

$$+ \left( -p, q_3 + p, q_0 + p, q_1 \right)j + \left( p, q_2 - p, q_1 + p, q_0 \right)k$$

If the quaternion is also pure, and we consider the two pure quaternions as vectors in $\mathbb{R}^3$, then the product is exactly a representation of the negation of the dot product of the two vectors and the cross product of the two vectors. Geometrically, the dot product is the projection of one vector into the axis of the other, while the cross product is the area of a parallelogram subtended by the two vectors and points in a direction normal to the parallelogram. The dot product is thus the component of the image pixel in the direction of the quaternion, and the cross product is a vector with direction related to the difference between the two quaternions. This linkage provides the rationale for the preferred extension of the Gabor filter to quaternions, as described in section 4.5.

### 3.4 Quaternionic Fourier Transforms

As mentioned earlier, Moxey, Sangwine and Ell have defined a series of Fourier transforms for quaternion-valued images (in [Moxey03] and [Sangw01]). Due to the non-commutative nature of the hypercomplex product, there are four specific Fourier transforms in their formulation: “right-handed” Fourier and inverse transforms, and “left-handed” Fourier and
inverse transforms. The “handed-ness” indicates which term appears first within the Fourier integral, the spatial function or the reducing exponential.

Pei et al (in [Pei01B]) also define a quaternionic Fourier transform, related to their definition of the quaternionic correlation. Again, their motivation in developing the Quaternionic Fourier Transform is to permit efficient implementation. They use a different nomenclature for the various transforms (types 1, 2, and 3 DFT), but the conclusions are generally similar.

The Fourier transform has value in providing understanding of the frequency behavior of signals and of systems. The quaternionic Fourier transform is similar in its meaning. To quote [Sangw01], "The spectrum is thus divided into quadrants. Any given pair of horizontal and vertical frequencies is represented in the spatial frequency domain by four quaternion values, one in each of the four quadrants." Another use of the Fourier transform in image processing is to increase the efficiency of certain image convolution operations; convolution in the spatial domain can be achieved by forming the product of the transforms of the functions. The quaternionic Fourier transform would offer the same efficiency advantage.

One interesting note from the work of Sangwine and Ell in this area is that their quaternionic Fourier transform is computed in a specific direction. An arbitrary vector in the quaternion space is used to compute the hypercomplex exponentials. They choose the vector that is diagonal to the unit color cube: \( \mu = \begin{bmatrix} \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \end{bmatrix} \). My definition of the quaternionic Gabor filter also uses an arbitrary vector in color space, but I will establish this vector in the direction most likely to produce good discriminability, as described in Chapter 5.

### 3.5 Convolutions using quaternionic color pixels

It has been observed that convolution of two quaternion-valued images or signals can be defined as follows, from [Moxey03].

\[
(f * g)(q, p) = \sum_{q=0}^{M-1} \sum_{p=0}^{N-1} f(q, p) g(q-m, p-n)^* 
\]  

(3-2)

This relationship should be considered as it relates to color pixels encoded as quaternions. First, since the quaternion product is non-commutative, the order of evaluation must in general
be carefully preserved. Secondly, if the color pixel \( f \) is a pure quaternion, as is discussed herein, and the convolution signal \( g \) is also pure, then multiplication is anti-commutative. Finally, a pure \( g \) will have a complex conjugate equal to its negation; \( g^* = -g \). Combining these results, we see that, for pure \( f \) and \( g \), a convenient symmetry exists in the convolution relation.

\[
f * g = \sum_{q=0}^{M-1} \sum_{p=0}^{N-1} f(q, p)(-g(q-m, p-n)) = \sum_{q=0}^{M-1} \sum_{p=0}^{N-1} g(q-m, p-n)f(q, p) \quad (3-3)
\]

For the purposes of feature extraction, the sign of the result is not significant, as long as it is consistent throughout. The test implementation used the form of convolution in equation 3-3b so as to obtain positive magnitudes for most jet components.

### 3.6 Extraction of quaternionic Gabor jets

The elastic graph method for face recognition uses feature vectors called jets. A jet is a collection of the responses of a number of related Gabor filters to the image at a particular location. In nearly all face recognition applications, the jet is composed of the responses of 40 filters. For each of eight equally-spaced orientation angles between 0 and \( \pi \), five filters are used whose frequencies are related by a ratio of \( \sqrt{2} \) (this ratio is carried over from the usual practice in hierarchical subsampled representations based on Gaussian filters; see [Crowl02]). Each filter's response is a complex number, resulting from the convolution of a complex filter with a real-valued image, so the jet is an 80-element vector.

For the quaternion case, we will adopt the same practice of using eight equally-spaced orientation angles and five scales related by \( \sqrt{2} \). Since the filter and image are both quaternion-valued, the convolution result will also be a quaternion with (generally) a non-zero real part. Therefore, the quaternion jets will have 160 scalar components. These jets characterize the image location in terms of not only frequency component and orientation but also in color, in terms of the projection of and the rotation angle between the region's color values and the given color axis \( \mu \).
3.7 Conclusions

To process color images in a systematic manner that takes advantage of the inherent relationships between the planes, we encode the color values as hypercomplex numbers. For color representations with three planes (RGB, HSI, etc.), the planes are assigned to the three complex portions of a quaternion-valued image. Generally, the real part remains at zero. This encoding provides a useful geometric property when the quaternionic product of a color image and a pure quaternion is formed. The result has a real part equal to the dot negative of the dot product of the color vector and the pure quaternion, while the imaginary part (itself a pure quaternion) is equal to the cross product of the two vectors. This provides information about the relative angles, rotations in space and magnitudes of the two vectors.
Chapter 4   Extension of Gabor Filter to Quaternions

4.1 Introduction

This chapter lays the groundwork for extension of the Gabor filter to $\mathbb{Q}$, the space of quaternions. There is surprisingly little theoretical guidance towards a “canonical” method of performing this extension. In the lack of strong theoretical approach, we adopt a practical approach. Just as the Fourier transform is highly useful because of the facility of the interpretation of its domain as frequency, we will look for formulations of the Gabor filter whose response has components that may interpreted in ways that help to understand and analyze the input signal.

In this chapter, the rationale for the form of the quaternionic Gabor filter is presented. Through the course of this research, four possible models were postulated; each one is described and the reasons for preferring the geometrically inspired model are discussed. In order to better understand the quaternionic Gabor filter, I present an examination of the fundamental properties of an orthogonal wavelet (zero-moments and orthogonality over integer scale and shift). This examination begins by describing the properties of the one-dimensional Gabor filter, then the two-dimensional complex Gabor filter and finally the quaternionic Gabor filter.

4.2 Four Possible Models

There are a number of possibilities for extending the Gabor filter to color images using the mathematical properties of quaternions. Four possible approaches were investigated to some degree. These are described below, along with the rationale for the choice made.

4.3 Direct Substitution of $\mu$ for $j$

If we identify a pure unit quaternion $q$ with the complex root $j$ in the real-valued 2-D Gabor filter, and substitute appropriately, based on the representation in [Wisko99] (refer to section 24), the following expression results:
\[ F_h(\bar{x}) = \frac{k_h^2}{\sigma^2} \exp \left( -\frac{k_h^2 \bar{x}^2}{2\sigma^2} \right) \left[ \exp \left( q \bar{k}_h \cdot \bar{x} \right) - \exp \left( -\frac{\sigma^2}{2} \right) \right] \] (4-1)

Applying the quaternion form of the Euler identity, the expression becomes

\[ F_h(\bar{x}) = \frac{k_h^2}{\sigma^2} \exp \left( -\frac{k_h^2 \bar{x}^2}{2\sigma^2} \right) \cos \left( \bar{k}_h \cdot \bar{x} \right) + q \sin \left( \bar{k}_h \cdot \bar{x} \right) - \exp \left( -\frac{\sigma^2}{2} \right) \] (4-2)

The first software implementations written in this research effort used this method of representing the Quaternionic Gabor filter. However, initial results were poor; filter responses oscillated even in regions where the image was stable. Some inspection revealed that the interactions between the real and complex parts of the Gabor filter were causing behavior similar to an underdamped response, in the form of a rotation in quaternion space. In addition, it was not possible to directly interpret the components of the responses. Thus, this alternative for extension of the Gabor filter was abandoned, and another alternative was sought.

### 4.4 Extension into Two Complex Axes

Ell (in [Ell92], §4.1) defines the Quaternionic Fourier Transform by use of two orthogonal complex exponentials:

\[
\hat{\mathcal{F}}[h(t,\tau)] := \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-j\omega t} h(t,\tau) e^{-j\nu\tau} dt d\tau = H[j\omega,k\nu]
\] (4-3)

and

\[
\hat{\mathcal{F}}^{-1}[H[j\omega,k\nu]] := \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{j\omega t} H[j\omega,k\nu] e^{j\nu\tau} d\nu d\omega = h(t,\tau)
\] (4-4)

This transform maps from real functions of two variables \(t\) and \(\tau\) to a complex functions of two variables \(\omega\) and \(\nu\) that may be interpreted as spatial frequencies of a sort. Ell shows that this QFT has scale and shift properties similar to the single-valued Fourier transform, and that it obeys a form of Parseval’s theorem.
In an attempt to extend frequency behavior of the Gabor filter to quaternions, we can define a Gabor filter using two complex roots, as Ell does for the Fourier transform. This results in the following expression:

\[
E_h(\vec{x}) = \frac{k_h^2}{\sigma^2}\exp\left(-\frac{k_h^2\vec{x}^2}{2\sigma^2}\right)\exp\left(i\vec{k}_h \cdot \vec{x}\right)\exp\left(j\vec{k}_h \cdot \vec{x}\right) - \exp\left(-\frac{\sigma^2}{2}\right)
\]  

\[
E_h(\vec{x}) = \frac{k_h^2}{\sigma^2}\exp\left(-\frac{k_h^2\vec{x}^2}{2\sigma^2}\right)\exp\left((i+j)\vec{k}_h \cdot \vec{x}\right) - \exp\left(-\frac{\sigma^2}{2}\right)
\]

The quaternion \((i+j)\) has a magnitude equal to \(\sqrt{2}\); factoring this out, we can obtain a pure unit quaternion (of a specific form, to be sure) and can use the Euler identity to yield:

\[
E_h(\vec{x}) = \frac{k_h^2}{\sigma^2}\exp\left(-\frac{k_h^2\vec{x}^2}{2\sigma^2}\right)\left[\frac{1}{\sqrt{2}}\left(\cos(\vec{k}_h \cdot \vec{x}) + (i+j)\sin(\vec{k}_h \cdot \vec{x})\right) - \exp\left(-\frac{\sigma^2}{2}\right)\right]
\]

This alternative was not implemented, for several reasons. This function has an oscillatory behavior parallel to the line \(i+j = 0\), and would therefore correlate strongly with sinusoidal signals in a particular direction in color space. There is no obvious criterion to select the color axes to identify with \(i\) and \(j\). Note that the fundamental wave direction is constrained to be perpendicular to the \(j\) axis. While this formulation does provide an insight into the frequency behavior of the color image, recognition features extracted in this fashion may not be expected to be well-matched to all colors in the spectrum. Chapter 5 describes in detail a method to determine the most relevant vector of projection for a collection of color images. It is desired to identify a method that will allow the use of such a vector. Extension of the complex Gabor into only two axes of \(\mathbb{Q}\) does not allow enough flexibility to accommodate a general vector of projection.

### 4.5 Geometrical Interpretation of the Product

The third possible form of the Quaternionic Gabor filter is based on two observations:

We are representing a color pixel as a pure quaternion, with the red, green and blue components assigned to the \(i, j\) and \(k\) axes;
The product of two pure quaternions, viewed as vectors in \( \mathbb{R}^3 \), has a very specific geometric interpretation: the real part of the result is the negative dot product of the two vectors, while the quaternion part is a vector equal to the cross product of the two vectors.

If we then define the Gabor filter as a pure quaternionic quantity, pointing in a particular “direction” in three-color space, then the resulting product contains two types of information: the (negation of the) projection of the color pixel onto the Gabor color axis (akin to an intensity), and the (negation of the) rotation that must be applied to the color pixel to make it parallel with the Gabor color axis (a form of hue information).

Note that three imaginary components of the resulting filter are in phase and differ only in magnitude. This does not seem like a very rich feature extraction filter, but the resulting intensity and hue information is very intuitively appealing.

One option to realize this form is to compose the product of a unit pure quaternion pointing in an “interesting” direction in three-color space, \( \mu \), with the real part of the usual Gabor filter. This form of the QGF can be written:

\[
F_h(x) = \mu \frac{k_h^2}{\sigma^2} \exp\left(-\frac{k_h^2 x^2}{2\sigma^2}\right) \left[ \cos(k_h \cdot x) - \exp\left(-\frac{\sigma^2}{2}\right) \right]
\]

(Basing the extension of the filter on a color axis of interest is also part of Ell’s quaternionic Fourier transform. In (4-8) it is the direction in color space against which the projection and the angle of color pixel vectors will be measured, in a sense. This is the form of the quaternionic Gabor filter that was implemented and used in the remainder of the work.

### 4.6 Opponent Color Feature Model

The fourth Quaternionic Gabor filter representation derives from an interpretation of human vision. Jain and Healey, in [Jain97], formulate a family of Gabor filters to mimic the operation of certain types of receptive fields in the retina. These receptive fields have multispectral responses of three types, as demonstrated below.
Figure 11 - Three types of receptive fields in the human retina (reproduced from [Jain97])

Type III is amply emulated by a Gabor filter on a single image plane, given an appropriate offset. Type II can be achieved by forming the difference in the response of a single filter to two different color planes.

Type I involves the difference of two Gabor filters at different scales, with opposite signs, applied to two different color planes. The resulting filter will have strong response to image locations where there is a localized difference in the color gradient, and the specific frequency of the Gabor filter will provide some characterization of the rapidity of the change. To replicate this characteristic, we need to convolve two color planes with Gabor filters of two different scales and form the difference. To build a comprehensive set of these filters would require a large number of different kernels, and would increase the length of the feature vectors dramatically. While this framework for filter definition was not chosen for further investigation, it is a possible field of future work.

4.7 Properties as a Filter

The quaternionic Gabor filter based on the geometric interpretation of the product is written in equation (4-8) above. It describes a quaternion-valued function that can be applied to images with quaternion values to extract features based on the frequency behavior of the color vectors. Changes in the direction of the color vector associated with linear features in the face
images will cause a peak in the response of one of the Gabor filters in the set used to extract
the jet – if the frequency or line width of the feature is within the range of spatial frequencies
covered by the filter scales used. In the application of the complex Gabor filters in the
published literature, the typical set of spatial frequencies used is stated as ranging from 16
pixels to 4 pixels for images of width 128x128, or sinusoidal periods of 0.25 to 0.0625 of the
width of the head (since most of the test images used in the published reports are about half
the width of the image.

In one way, a set of Gabor filters can be thought of as a variation on the discrete Fourier
transform. In fact, in his original paper [Gabor46] Dennis Gabor went on to deal with the
short-time Fourier transform. In digital signal processing, signals that are processed using a
discrete Fourier transform are generally first restricted in support by the application of a
suitable windowing function, to retain the temporal region of interest without causing ringing
at the ends of the range (the Gibbs phenomenon). A Gabor filter in one dimension is quite
similar to successive application of a Gaussian window and a single sinusoidal convolution at
the center point. This observation raises two questions:

Gabor’s selection of the Gaussian envelope was done to minimize the simultaneous spectral
and temporal uncertainties. What if another criterion is more suitable for use in feature
extraction? This could mean that another set of functions, composed of complex exponentials
modulated by (possibly) one of the other common signal processing windowing functions
such as Chebyshev or Parzen, may be superior.

Are there other alternatives for extending a one-dimensional Gabor filter to two dimensions?
The notion of rotating the planar Gabor function to the desired angle is logical, but may not be
optimal.

4.8 Properties as a "Wavelet"

The one-dimensional Gabor filter is commonly called a wavelet. However, in order for a
function to qualify as a wavelet (as defined by Daubechies and others), it must satisfy several
key properties:

- The value of the wavelet function at $t=0$ must be 1;
• The 0th moment must vanish

• All upper moments must vanish

• The function must be orthogonal with itself over integer shift

• The function must be orthogonal with itself over integer scale

In addition, wavelets that meet certain other properties are noteworthy:

• Compact support

• Symmetry

• Regularity (infinite differentiability)

The one-dimensional Gabor function, as described by Gabor as

$$\varphi(t) = e^{-\alpha^2(t-t_0)^2} e^{j(2\pi f_0 t + \phi)}$$  \hspace{1cm} (4-9)

may be evaluated against each of the properties of a wavelet.

Evaluate at the origin:

$$\left. e^{-\alpha^2(t-t_0)^2 + j(2\pi f_0 t_0 + \phi)} \right|_{t=t_0} = e^{-\alpha^2(0)^2 + j(2\pi f_0 t_0 + \phi)} = e^0 e^{j(0)}$$ \hspace{1cm} (4-10)

$$e^0 e^{j(0)} = 1, \text{ with } \phi = -2\pi f_0 t_0$$

The 0th moment:

$$\int_{-\infty}^{\infty} \varphi(t) \, dt = \int_{-\infty}^{\infty} e^{-\alpha^2(t-t_0)^2 + j(2\pi f_0 t + \phi)} \, dt =$$

$$\int_{0}^{\infty} e^{-\alpha^2(t-t_0)^2} e^{j(2\pi f_0 t + \phi)} \, dt = \int_{0}^{\infty} e^{-\alpha^2(t-t_0)^2} e^{j(2\pi f_0 t + \phi)} \, dt = 0$$ \hspace{1cm} (4-11)

The nth moment:

$$\int_{-\infty}^{\infty} \varphi^n(t) \, dt = \int_{-\infty}^{\infty} e^{-n\alpha^2(t-t_0)^2 + jn(2\pi f_0 t + \phi)} \, dt =$$ \hspace{1cm} (4-12)
\[ \int_{0}^{\infty} e^{-\alpha t^2} j_{2\pi f_{0} + \phi} dt - \int_{-\infty}^{0} e^{-\alpha t^2} j_{2\pi f_{0} + \phi} dt = 0 \]

Orthogonality over shift:

\[ \int_{-\infty}^{\infty} \phi(t) \phi(t-\tau) dt = \int_{-\infty}^{\infty} e^{-\alpha^2 (t-\tau)^2 + j(2\pi f_{0}(t-\tau))} e^{-\alpha^2 (t-t_0)^2 + j(2\pi f_{0} + \phi)} dt = \]

\[ \int_{-\infty}^{\infty} e^{-\alpha^2 (t-\tau)^2 - \alpha^2 (t-t_0)^2 + j(2\pi f_{0}(t-\tau)) + j(2\pi f_{0} + \phi)} dt = 0 \quad (4-13) \]

Orthogonality over scale:

\[ \int_{-\infty}^{\infty} \phi(t) \phi(\beta t) dt = \int_{-\infty}^{\infty} e^{-\alpha^2 (t-t_0)^2 + j(2\pi f_{0}(t-t_0))} e^{-\alpha^2 (\beta t-t_0)^2 + j(2\pi f_{0}(\beta t-t_0))} dt = \]

\[ \int_{-\infty}^{\infty} e^{-\alpha^2 ((1+\beta)t-2t_0)^2 + j(2\pi f_{0}((1+\beta)t-2t_0))} dt = \int_{-\infty}^{\infty} e^{-\alpha^2 (t')^2 + j(2\pi f_{0}(t'))} dt' = \]

\[ \int_{0}^{\infty} e^{-\alpha^2 (t')^2 + j(2\pi f_{0}(t'))} dt' - \int_{-\infty}^{0} e^{-\alpha^2 (t')^2 + j(2\pi f_{0}(t'))} dt' = 0, \text{ from (4-11) above.} \quad (4-14) \]

Thus, the one-dimensional Gabor function is a wavelet with infinite support, odd symmetry in its complex part and even symmetry in its real part, and is entirely regular since it is infinitely differentiable.

The two-dimensional Gabor function does not meet all of these conditions. The magnitude of the wave at the origin is zero, and the moments vanish, but it is not self-orthogonal. To see this, examine the same set of conditions for Wiskott's formulation of the two-dimensional Gabor function.

Evaluate at the origin:

\[ \int_{0}^{\infty} e^{-\alpha^2 (t-t_0)^2 + j(2\pi f_{0}(t-t_0))} dt - \int_{-\infty}^{0} e^{-\alpha^2 (t-t_0)^2 + j(2\pi f_{0} + \phi)} dt = 0 \quad (4-15) \]
\[ \left| \frac{k^2_{\mu,\nu}}{\sigma^2} \exp \left( -\frac{k^2_{\mu,\nu} (x^2 + y^2)}{2\sigma^2} \right) \right| \| \exp \left( jk_{\mu,\nu} \cdot (x, y) \right) - \exp \left( -\frac{\sigma^2}{2} \right) \|_{y=0, j=0} = \]

\[ \left| \frac{k^2_{\mu,\nu}}{\sigma^2} \exp \left( -\frac{k^2_{\mu,\nu} (0)}{2\sigma^2} \right) \right| \| \exp (0) - \exp \left( -\frac{\sigma^2}{2} \right) \|, \text{with } \sigma = 2\pi, k_{\mu,\nu} = \frac{k_{\text{max}}}{f_v} e^{j\frac{\pi}{\psi}}, \]

\[ \left| \exp \left( j\frac{\mu}{4} \right) \frac{k_{\text{max}}^2}{4\pi^2} \left[ 1 - \exp (-2\pi^2) \right] \right| = \left| \frac{k_{\text{max}}^2}{4\pi^2} \left[ \exp \left( j\frac{\mu}{4} \right) - \exp \left( j\frac{\mu}{4} - 2\pi^2 \right) \right] \right| = \]

\[ \left| \frac{k_{\text{max}}}{4\pi^2} \left[ \exp \left( j\frac{\mu}{4} \right) - \exp \left( j\frac{\mu}{4} - 2\pi^2 \right) \right] \right| = \left( \frac{k_{\text{max}}}{2\pi} \right)^2 = 1, \text{ for } k_{\text{max}} = 2\pi \]

The 0th moment: \[ \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi(x, y) \, dx \, dy = \]

\[ \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{k^2_{\gamma}}{\sigma^2} \exp \left( -\frac{k^2_{\gamma} (x^2 + y^2)}{2\sigma^2} \right) \left[ \exp \left( jk_{\gamma} \cdot (x, y) \right) - \exp \left( -\frac{\sigma^2}{2} \right) \right] \, dx \, dy = 0 \]  \quad (4-16)

The nth moment: \[ \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi^n(x, y) \, dx \, dy = \]

\[ \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{k^{2n}}{\sigma^{2n}} \exp \left( -nk_{\gamma} \frac{(x^2 + y^2)}{2\sigma^2} \right) \left[ \exp \left( jnk_{\gamma} \cdot (x, y) \right) - \exp \left( -\frac{n\sigma^2}{2} \right) \right] \, dx \, dy = 0 \]  \quad (4-17)

However, the two-dimensional Gabor function is not self-orthogonal for integer shifts:

\[ \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi(x, y) \, dx \, dy = \]

\[ \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{k^2_{\gamma}}{\sigma^2} \exp \left( -\frac{k^2_{\gamma} ((x-\kappa)^2 + (y-\lambda)^2 + (x^2 + y^2))}{2\sigma^2} \right) \, dx \, dy = 0 \]  \quad (4-18)
\[
\begin{align*}
\left[ \exp \left( jk_h \begin{bmatrix} x - \kappa \\ y - \lambda \end{bmatrix} \right) - \exp \left( -\frac{\sigma^2}{2} \right) \right] \left[ \exp \left( jk_h \begin{bmatrix} x \\ y \end{bmatrix} \right) - \exp \left( -\frac{\sigma^2}{2} \right) \right] & dxdy \\
= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{k_h^2}{\sigma^2} \exp \left( -\frac{k_h^2}{\sigma^2} \left( 2x^2 - 2\kappa x + \kappa^2 + 2y^2 - 2\lambda y + \lambda^2 \right) \right) & \\
& \left[ \exp \left( j \left( k_{hx} (x - \kappa) + k_{hy} (y - \lambda) \right) \right) - \exp \left( -\frac{\sigma^2}{2} \right) \right] \left[ \exp \left( j \left( k_{hx} x + k_{hy} y \right) \right) - \exp \left( -\frac{\sigma^2}{2} \right) \right] dxdy \\
= \exp \left( -\frac{k_h^2}{2\sigma^2} \right) \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \exp \left( -\frac{k_h^2}{\sigma^2} \left( \frac{\kappa^2}{\sigma^2} - \kappa \right) \right) & \\
& \left[ \exp \left( j \left( k_{hx} (2x - \kappa) + k_{hy} (2y - \lambda) \right) \right) - \exp \left( j \left( k_{hx} x + k_{hy} y \right) - \frac{\sigma^2}{2} \right) \right] \left[ \exp \left( j \left( k_{hx} x + k_{hy} y \right) - \frac{\sigma^2}{2} \right) \right] dxdy \\
& - \exp \left( j \left( k_{hx} (2x - \kappa) + k_{hy} (2y - \lambda) \right) - \frac{\sigma^2}{2} \right) + \exp \left( -\frac{\sigma^2}{2} \right) \\
\end{align*}
\]

It is sufficient to show that the integral is non-zero for a single value of shift; choose \( \kappa = 1 \) and \( \lambda = 1 \).

Define a factor \( K = \frac{k_h^2}{\sigma^2} \exp \left( -\frac{k_h^2}{2\sigma^2} \left( \frac{1 + \lambda^2}{\sigma^2} \right) \right) \):

\[
\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \varphi(x, y) \varphi(x - \kappa, y - \lambda) dxdy = K \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \exp \left( -\frac{k_h^2}{\sigma^2} \left( \frac{x^2 - x + y^2 - y}{\sigma^2} \right) \right)
\]

\[
\left[ \exp \left( j \left( k_{hx} (2x - 1) + k_{hy} (2y - 1) \right) \right) - \exp \left( j \left( k_{hx} x + k_{hy} y \right) - \frac{\sigma^2}{2} \right) \right] \\
- \exp \left( j \left( k_{hx} (2x - 1) + k_{hy} (2y - 1) \right) - \frac{\sigma^2}{2} \right) + \exp \left( -\frac{\sigma^2}{2} \right) \\
\right] dxdy
\]
So, the two-dimensional Gabor filter is not an orthogonal wavelet, despite the frequent application of the term "wavelet" to it. Similarly, the quaternionic Gabor filter is not a wavelet as usually defined, since the function is not orthogonal to integer shifts in any axis. Demonstration of this fact is not included in this dissertation, but can be most easily verified numerically.

This has the important implication that the coefficients of expansion of any image in terms of a family of Gabor function cannot be computed by simply convolving the image with the Gabor functions one at a time. Since the basis formed by the two-dimensional Gabor functions is not orthogonal, the mixed products are non-zero. This property is well-explored in [Daugm88]. While this complicates the use of Gabor functions for image compression and representation, it has no direct bearing on their use for feature extraction. However, it must be remembered that the extension of the one-dimensional Gabor elementary function into two dimensions involved the loss of orthogonality, and this is also true for the quaternionic Gabor filter as it is defined herein.

It can be observed that my formulation of the QGF has a norm of 1 at its center.

$$F_h(\mathbf{x})_|_{\mathbf{x} = 0} = \mu \frac{k_h^2}{\sigma^2} \exp \left( -\frac{k_h^2 \mathbf{x}^2}{2} \right) \left[ \cos \left( \mathbf{k}_h \cdot \mathbf{x} \right) - \exp \left( -\frac{\sigma^2}{2} \right) \right]$$

(4-19)
\[
\mu \frac{k_h^2}{\sigma^2} \exp(0) \left[ \cos(0) - \exp\left(-\frac{\sigma^2}{2}\right) \right] = \mu \frac{k_h^2}{4\pi^2} \left[ 1 - \exp\left(-2\pi^2\right) \right] = \mu
\]

\[\|F_h(\bar{x})\|_{d=0} = \|\mu\| = 1, \text{ by the definition of } \mu \text{ in section 4.5 above.}\]

In addition, the QGF integrated across its domain produces the zero vector (thus, it has "no DC bias"). This can be verified by computing the 0th moment.

\[
\int_{-\infty}^{\infty} F_h(\bar{x}) d\bar{x} = \int_{-\infty}^{\infty} \mu \frac{k_h^2}{\sigma^2} \exp\left(-\frac{k_h^2 \bar{x}^2}{2\sigma^2}\right) \left[ \cos(k_h \cdot \bar{x}) - \exp\left(-\frac{\sigma^2}{2}\right) \right] d\bar{x} = 0 \quad (4-20)
\]

Verification of this analytically requires a full definition of quaternion integration over the domain of vector arguments, which has not been done. Alternatively, the integral has been approximated by the use of summations over the spatial domain, and it has been verified that the result is a quaternion whose components are zero to the accuracy of the implementation.

Specific application of the implications of these properties is left for future work. However, it is important to note that the quaternionic Gabor filter is not an orthogonal wavelet – a fact that it shares with the complex two-dimensional Gabor filter (as observed by Daugman in [Daugm88]). For the purposes of this research, the key element is the use of the filter responses to various regions of color face images. These responses are concatenated into feature vectors and used for recognition. It will be seen from the results of testing that these features do provide useful characterizations of the face image, irrespective of the fact that the underlying filters are not orthogonal.

### 4.9 Conclusions

This chapter has introduced an extension of the Gabor filter to quaternions. Four possible methods of extension have been presented: direct substitution of a quaternion \( q \) for the imaginary root \( j \), extension into two of the three complex axes, substitution of a pure quaternion to obtain a logical geometric interpretation of the result, and use of a color opponent Gabor model developed by Jain and Healey. Based on initial experiments and
analysis, the geometric model was chosen for the remainder of this work. However, the color opponent model also has the potential for useful behavior.

The quaternionic Gabor filter as defined here (occasionally abbreviated as $QGF$ in this report) has several properties similar to those of one-dimensional and complex two-dimensional Gabor filters. It has a maximum norm of 1 at its center and it integrates to a 0 quaternion over its domain. However, it is not a classical wavelet since it is not self-orthogonal to integer shifts in the spatial domain. This is also the case for the complex two-dimensional Gabor filter, and only provides a complication in the use of the Gabor filter for image compression and representation. While the fact that the quaternionic Gabor filter is not an orthogonal wavelet is important for consideration of its use in representing an image (for color image compression, for example), it does not weaken the use of the filter for feature extraction purposes.
Chapter 5  Optimal Color Projection for Color Images

5.1 Introduction

In Chapter 4, the quaternionic Gabor filter was formulated. The definition chosen contains within it an arbitrary unit pure quaternion $\mu$, which is described as a vector pointing in an "interesting" direction in the color space. This general formulation is adequate for initial discussion, but specific application of the filter requires a selection of $\mu$. It is evident that selection of a vector onto which the color pixels will project a very small range is not optimal; we adopt the opposite conclusion: that a suitable choice for $\mu$ is one pertaining the color vector onto which the face images project onto a large range – or that the projection contains the most useful information in some sense. As mentioned earlier, Moxey et al. [Moxey03] make an arbitrary selection of the diagonal of the unit color cube for an analogous color vector in their quaternionic Fourier transform. However, there is an opportunity to choose the color axis that will optimize performance of the resulting filters.

This chapter discusses a strategy for analytic selection of the optimal color axis for projection of a collection of images. The initial discussion relates to the more general application of conversion of color imagery to monochrome in a manner that is best suited for a particular application. The following sections describe the identification of the optimal color projection axis in two ways: determination of the principal components and minimization of the square error from the line. Results from both methods are assessed by evaluation of the eigenvalues of the sample images produced by each, and by measuring the effectiveness of a complete eigenface implementation operating on the resulting images.

5.2 Background and Related Work

Most single-view face recognition systems operate using intensity (monochromatic) information alone. This is true even for systems that accept color imagery as input. The reason for this is not that multispectral data is lacking in information content, but often because of practical considerations – difficulties associated with illumination and color balancing, for example, as well as the additional cost of color imagers. Associated with this is
a lack of color image databases with which to develop and test new algorithms. Although work is in progress to solve the color constancy problem (e.g., \cite{Finla01}), those efforts are still in the research stage.

When color information is present, most face-recognition systems convert the color information to monochromatic form using simple transformations. For example, a common mapping \cite{Gonza92, Hunt99} produces an intensity value \( I_i \) by taking the average of red, green, and blue (RGB) values \( I_r, I_g, \) and \( I_b \) respectively:

\[
I_i(x, y) = \frac{I_r(x, y) + I_g(x, y) + I_b(x, y)}{3}
\]

The resulting image is then used for feature extraction and analysis.

More effective system performance is possible if a color transformation is chosen that better matches the task at hand. For example, the mapping in (5-1) implicitly assumes a uniform distribution of color values over the entire color space. For a task such as face recognition, color values tend to be more tightly confined to a small portion of the color space, and it is possible to exploit this narrow concentration during color conversion. If the transformation is selected based on the expected color distribution, then it is reasonable to expect improved recognition accuracies.

We assume that frontal color views of the human face are available, and we develop a method for selecting alternate weightings of the separate color values in computing a single monochromatic value. Given the rich color content of the human face, it is desirable to maximize use of this content even when full-color computation and matching is not used. As an illustration of this framework, we have used the Karhunen-Loève (K-L) transformation (also known as principal components analysis) of observed distributions in the color space to determine improved the mapping.

Other work \cite{Albio01} has shown that alternative color spaces provide no real benefit for skin detection, since they do not increase the separability of the skin and non-skin classes. However, to extract features for face recognition, we do not wish to discriminate skin from non-skin regions, but rather to extract meaningful image features within the skin area.
Chapter 5 – Optimal Color Projection for Color Images

Queisser [Queis97] used the properties of color distributions of a set of similar images to select a new color space for object classification. Abbott and Zhao [Abbot96], [Zhao96] developed a color-space quantization approach for the recognition of naturally textured objects, but did not consider that for face recognition. Heseltine et al [Hesel02] measured the performance effect on eigenface-based face recognition of a number of preprocessing techniques, including several color transformations (RGB to Hue, brightness-insensitive Hue, etc.) and found that the color pre-processes evaluated actually degraded the recognition accuracy. However, the techniques that they explored were general color transformations that were not based on the content of the images.

5.3 K-L Analysis

Pixels in the original color image can be represented as the vector

$$I(x, y) = [I_r(x, y) \ I_g(x, y) \ I_b(x, y)]^T$$

(5-2)

where the r, g, and b subscripts denote the red, green, and blue color planes, respectively. Face recognition systems typically use an intensity plane composed of equal proportions of the red, green and blue planes, computed using (5-1) above. Human face images exhibit common characteristics that can be exploited in the conversion from a full-color representation to a monochrome image. In the hue-saturation plane, for example, face pixels from a mixture of ethnic groups are well-clustered [Gong00], with only the intensity plane varying markedly. This suggests that the standard intensity plane is in fact more sensitive to variation due to ethnic type, which is undesirable.

To determine an improved linear transformation, we want to find the optimum transformation vector \( w \) such that

$$M(x, y) = w^T I(x, y)$$

(5-3)

where \( I \) is the original color image and \( M \) is the resulting single-plane image. We make the assumption that the optimum transformation corresponds closely to the expected distribution of pixel values within the original color space. With this in mind, it is possible to select \( w \) by using the Karhunen-Loève transformation to determine the projection with uncorrelated axes. The resulting color space has been called the “Karhunen-Loève color space” for an
unspecified pixel population [Lee96]; here, we specifically restrict it to the face area. For a
given distribution of pixel values, the eigenvector corresponding to the largest eigenvalue
defines the direction along which the data is the least correlated, and therefore most likely to
be of use in recognition tasks.

The K-L transformation is determined from the covariance matrix of the distribution. For this
application, the input datum is the ensemble of pixel values from a set of training images,
taken from the region containing the face. We form the covariance matrix $S$,

$$ S = \frac{1}{M} \begin{bmatrix} \sum_{m} p_r^2 & \sum_{m} p_r p_g & \sum_{m} p_r p_b \\ \sum_{m} p_g p_r & \sum_{m} p_g^2 & \sum_{m} p_g p_b \\ \sum_{m} p_b p_r & \sum_{m} p_b p_g & \sum_{m} p_b^2 \end{bmatrix} - \frac{1}{M^2} \begin{bmatrix} \sum_{m} p_r \\ \sum_{m} p_g \\ \sum_{m} p_b \end{bmatrix} \begin{bmatrix} \sum_{m} p_r \\ \sum_{m} p_g \\ \sum_{m} p_b \end{bmatrix}^T \quad (5-4) $$

where $p$ is the collection of $M$ color pixel vectors. The K-L transformation is then given by
the eigenvectors $\{u_i\}$ of $S$, concatenated into the matrix $U = [u_1 \ u_2 \ u_3]$. The eigenvector $u_1$, associated with the largest eigenvalue, is of primary interest here; it represents the direction of
most variability in the data within the original space. Projection of RGB values onto this axis
represents a color-to-grayscale conversion with the highest potential for discrimination.

The normalization of the conversion vector $w$ requires consideration. A unit vector will, by
definition, not change the magnitude of the vector quantity that it operates on. However, this
is not appropriate for conversion of three-component color quantities (where each component
can range up to full-scale) to monochrome, since any three-color vector with magnitude
greater than unity will saturate in the monochrome plane. Saturation is prevented by
normalizing the vector having RGB components at full scale, to a magnitude of 1. Therefore,
the conversion vectors are normalized by $\sqrt{3}$. 
The images used in this study are frontal-view, color face images from the Ljubljana and Essex databases ([Ljub02], [Space96]). Each image is of size 240 rows by 300 columns. Prior to this study, the images were spatially registered so that the centers of the eye sockets are at fixed locations, the line between the eye centers is horizontal, and the distance between eye centers is 60 pixels, in accordance with developing standards for face recognition image interchange [Griff02]. No effort was made to color-correct or contrast-equalize the images.

To determine the color conversion that is most suited for the face features, we process only a portion of the face image that represents the area of the face with minimal included background and hair. The extent to be processed, a region 90 pixels wide by 140 pixels high, is indicated in Figure 12. This smaller image extent was chosen to eliminate effects from undesirable artifacts such as hair, background, strong highlights on the left side of the face and the collar of clothing; all of these regions contained colors that were not relevant to the analysis.
5.3.1 Analysis of RGB Color Images

The Karhunen-Loève analysis yields an eigenvector $u_i$ describing the axis of projection with the largest variance in the original data, which we call the conversion vector. Let the 3 components of this vector be represented by

$$u_i = [u_{i1} \ u_{i2} \ u_{i3}]^T$$  \hspace{1cm} (5-5)

Because this vector has unit length $r$:

$$r = \sqrt{u_{i1}^2 + u_{i2}^2 + u_{i3}^2} = 1$$  \hspace{1cm} (5-6)

we can represent it using spherical coordinates and completely describe the color mapping by the two angular quantities $\theta$ and $\phi$:

$$\phi = \arccos(u_{i3}) \quad \theta = \arccos\left(\frac{u_{i1}}{\sin(\phi)}\right)$$  \hspace{1cm} (5-7)

To illustrate the meaningfulness of the transformation, several scatter diagrams are shown in Figure 13. Four collections of face images are represented, as well as some natural images of random content. For each image, the color histogram was computed and conversion vector $u_i$ obtained. The resulting conversion vectors are indicated as points in $[\phi, \theta]$ space. Each face image collection consists of several sets of 21 images each for a single individual. The natural images contain a mix of object types, including landscapes, photographs of sporting events, and astronomy images.
Figure 13 - Principal component directions in spherical coordinates for several histograms in RGB space

It can be seen that the optimal color conversion vectors $u_i$ computed for the face images are distinct from those for more general natural images, indicating that the red, green and blue color planes carry different degrees of information for the specific class of face images. The figure also indicates the position in this space of an equal-weighted color conversion, which appears to represent a good estimate for the optimal conversion for general natural images, but is not as well suited for the face image collections. The selection of face databases used in our testing contain color distributions that generally correspond to $[\phi = 1.01, \theta = 0.662]$, which in turn corresponds to a conversion vector of

$$u_i = \frac{1}{\sqrt{3}} \begin{bmatrix} \sin \phi \cos \theta \\ \sin \phi \sin \theta \\ \cos \phi \end{bmatrix} = \begin{bmatrix} 0.3858 \\ 0.3004 \\ 0.3070 \end{bmatrix}$$

(5-8)

This should be compared with the equal-weighted values of

$$\begin{bmatrix} 0.333 \\ 0.333 \\ 0.333 \end{bmatrix}$$

(5-9)
Note that the K-L procedure, for these images, results in a color space that more heavily weights the red color component than the green and blue. This indicates that face images contain more uncorrelated variation in the red plane than in the green or blue planes.

RGB is not always the most convenient space in which to process color information. The CIE tristimulus system represents a color in terms of its three coordinates relative to a reference color, usually a standard illuminant [MacAd85]. However, equal distances in the XYZ space are perceived as unequal, so the L-a-b color space is defined so that color distances are perceived as linear.

### 5.3.2 Analysis of L-a-b Color Images

The L-a-b space is defined as follows [MacAd85]:

\[
p_L = 116 \left( \frac{p_r}{Y_0} \right)^{\frac{1}{3}} - 16 \\
p_a = 500 \left[ \left( \frac{p_r}{X_0} \right)^{\frac{1}{3}} - \left( \frac{p_r}{Y_0} \right)^{\frac{1}{3}} \right] \\
p_b = 200 \left[ \left( \frac{p_r}{Y_0} \right)^{\frac{1}{3}} - \left( \frac{p_r}{Z_0} \right)^{\frac{1}{3}} \right]
\]

where

\[
\begin{bmatrix}
p_x \\
p_y \\
p_z
\end{bmatrix} =
\begin{bmatrix}
0.412453 & 0.357580 & 0.189423 \\
0.212671 & 0.715160 & 0.072169 \\
0.019334 & 0.119193 & 0.950227
\end{bmatrix}
\begin{bmatrix}
p_r \\
p_g \\
p_b
\end{bmatrix}
\]

for the D65 standard illuminant used as the color reference point:

\[
\begin{bmatrix}
X_0 & Y_0 & Z_0
\end{bmatrix} =
\begin{bmatrix}
1004.26 & 1056.79 & 1150.71
\end{bmatrix}
\]

This K-L-based approach for selecting the color conversion produces a linear transformation of the RGB color values; thus, we could expect that using the K-L process on the XYZ values would produce the same result to within computation accuracy. However, the relation
between RGB and L-a-b is nonlinear, and the L-a-b space is in some sense more relevant to human perception, so that application of the K-L procedure defined in the previous section would be expected to produce useful results.

In fact, as can be seen in Figure 14, the K-L transformation on L-a-b data does not yield distinctive data for face pixels as opposed to image pixels from more general scenes. This suggested that the “optimal” color conversion obtained from L-a-b data does not provide any beneficial added feature content. Experimentation with the eigenvalues of face images converted to L-a-b representation, and then projected onto the axis found by using K-L on the resulting histogram data (as described below) showed that this was the case; information contained in the most significant n axes was no greater (and in fact frequently less) than for the L plane of the corresponding L-a-b images. It is possible that a transformation resulting in a linear perception of color distance inherently concentrates useful detail information in the L plane.

![Figure 14 - Principal component directions in spherical coordinates for several histograms in L-a-b space](image-url)
5.4 Line-fit Analysis

Queisser discusses (in [Queis97]) the use of a least-squared-error line fit to RGB data to define a new color axis that is best suited to images of a particular class of object. In his study, images of wood panels and food products were shown to be more suited for object detection and inspection in the resulting single-color plane than in any of the HSI axes. The other axes relate to additional magnitude and chromaticity information.

Consider a similar approach in the RGB space. A least-squared-error fit is performed to the RGB data, with the added constraint that the new axis of projection should pass through the RGB origin. The purpose of this is to force a pixel with zero in all color planes to map to a black pixel in the new space. The transformation matrix is as follows:

\[
\begin{bmatrix}
\beta \\
s \\
t
\end{bmatrix} =
\begin{bmatrix}
\frac{\bar{R}}{\sqrt{\bar{R}^2 + \bar{G}^2}} & \frac{\bar{G}}{\sqrt{\bar{R}^2 + \bar{G}^2}} & 0 \\
\frac{\bar{R}b}{\sqrt{\bar{R}^2 + \bar{G}^2}} & \frac{\bar{G}b}{\sqrt{\bar{R}^2 + \bar{G}^2}} & \frac{\bar{R}^2 + \bar{G}^2}{\sqrt{\bar{R}^2 + \bar{G}^2}}
\end{bmatrix}
\begin{bmatrix}
R \\
G \\
B
\end{bmatrix}
\]

(5-15)

Applying (14) to the pixel data from the face box areas of the sample databases, we obtain the data presented in Figure 14 as the “line fit” data. As before, only the primary axis is of interest, \( \beta \) in this transformation. The results are very similar to those obtained by the K-L method, but with much lower computational cost because only the red, green and blue sample means are required.

5.5 Eigenface Performance comparison

To evaluate the effect of our color conversion method on face recognition accuracy, it is possible to consider the effect on performance of the well-known eigenface method [Turk91, Groth99]. This technique uses principal components analysis of a collection of face images, treated as one-dimensional vectors, to determine the linear combinations of pixel locations that form the best projective axes for the collection. Early work in this area focused on the use of a small set of these projections to adequately represent a face image, while later work (beginning around 1990) applied this same technique to recognition. The new “face space”
defined by the $M$ most significant basis vectors, called “eigenfaces”, is used for pattern recognition based on a distance measure.

For any principal component analysis, the ratio of an eigenvalue to the sum of all the eigenvalues is proportional to the mean squared error implied by exclusion of the corresponding eigenvector [Nadle93]. Thus, we can examine the cumulative sum of eigenvalues 1 through $n$, plotted versus $n$, to compare the information contained in the first $n$ eigenfaces (the “principal components”). In this way, it is possible to predict the performance of the eigenface method on the two databases. Table 3 shows the individual and cumulative eigenvalues for a typical database of face images.

Table 3 - Eigenvalues for a typical face database using K-L to determine the RGB conversion

<table>
<thead>
<tr>
<th>index</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>equal weighted RGB conversion</td>
<td>eigenvalue $\lambda$</td>
<td>0.51613</td>
<td>0.13616</td>
<td>0.06179</td>
<td>0.05184</td>
<td>0.04326</td>
<td>0.03875</td>
<td>0.02879</td>
</tr>
<tr>
<td></td>
<td>cumulative $\Sigma \lambda$</td>
<td>0.51613</td>
<td>0.65229</td>
<td>0.71408</td>
<td>0.76592</td>
<td>0.80918</td>
<td>0.84793</td>
<td>0.87672</td>
</tr>
<tr>
<td>K-L computed RGB conversion</td>
<td>eigenvalue $\lambda$</td>
<td>0.53441</td>
<td>0.13210</td>
<td>0.05929</td>
<td>0.05223</td>
<td>0.04181</td>
<td>0.03812</td>
<td>0.02850</td>
</tr>
<tr>
<td></td>
<td>cumulative $\Sigma \lambda$</td>
<td>0.53280</td>
<td>0.66651</td>
<td>0.72580</td>
<td>0.77803</td>
<td>0.81984</td>
<td>0.85796</td>
<td>0.88646</td>
</tr>
</tbody>
</table>

Figure 15 shows a plot of the cumulative eigenvalues, which gives a measure of the accuracy achievable by truncating all higher eigenvalues. Using the optimized color conversion produces a modest, yet consistent, improvement in the potential accuracy. The increased information contained is more pronounced for the more significant eigenvalues.
By comparison, we also evaluated the magnitude of the initial eigenvectors for the eigenface method when using the line-fit method described in Section V. The cumulative eigenvalues computed by using the $\beta$ axis as the new image plane are shown in Table 4, and exhibit a similar increase in information in the lowest eigenfaces. In fact, for all of the databases we examined, the use of the line-fit gave essentially equal performance as measured by the normalized eigenvalues.

Table 4 - Eigenvalues for a typical face database using the Line Fit method to determine RGB conversion

<table>
<thead>
<tr>
<th></th>
<th>index</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>equal weighted</td>
<td>eigenvalue $\lambda$</td>
<td>0.51613</td>
<td>0.13616</td>
<td>0.06179</td>
<td>0.05184</td>
<td>0.04326</td>
<td>0.03875</td>
<td>0.02879</td>
<td>0.02683</td>
</tr>
<tr>
<td></td>
<td>cumulative $\Sigma \lambda$</td>
<td>0.51613</td>
<td>0.65229</td>
<td>0.71408</td>
<td>0.76592</td>
<td>0.80918</td>
<td>0.84793</td>
<td>0.87672</td>
<td>0.90354</td>
</tr>
<tr>
<td>line fit</td>
<td>eigenvalue $\lambda$</td>
<td>0.53280</td>
<td>0.13175</td>
<td>0.05801</td>
<td>0.05210</td>
<td>0.04064</td>
<td>0.03644</td>
<td>0.02898</td>
<td>0.02512</td>
</tr>
<tr>
<td></td>
<td>cumulative $\Sigma \lambda$</td>
<td>0.53280</td>
<td>0.66445</td>
<td>0.72256</td>
<td>0.77467</td>
<td>0.81531</td>
<td>0.85174</td>
<td>0.88073</td>
<td>0.90585</td>
</tr>
</tbody>
</table>

For confirmation of these predictions of increased performance, we measured the face recognition accuracy on a complete eigenface recognition implementation. For our test, a training phase and a test phase were implemented. The training phase computes the desired transformation by solving for the eigenvalues of the matrix composed of the concatenation of the training images. Testing is performed by applying this transformation to a set of probe
images of the same individuals, and measuring the Euclidean distance from the probe image data to the exemplars of each individual, defined as the average in “face space” of each training image of that individual. The probe images were not present in the training set. Note that the eigenface implementation was fairly simplistic; our objective was not to achieve high recognition accuracy overall but to measure the effect of using our color conversion.

To measure the performance in a consistent fashion, we adopted the method used in the NIST FERET studies ([Phill00]). The results for each probe image are ranked in order of increasing Euclidean distance. The performance score for a particular experiment, $R_n$, is defined to be the ratio of the number of times that the correct identity is in the top $n$ candidates (the $n$ nearest exemplars to the probe image) to the total number of probe images tested.

Table 5 summarizes the eigenface performance for three values of $n$ (2, 5, and 10) for a particularly difficult database of 280 images. Many of the images exhibit poor contrast, and there is significant variation in expression by the human subjects. Two sets of results are shown, the first for a typical equal-weighted conversion from RGB to monochrome, and the second for a transformation vector derived using the K-L procedure described above. The results show significant improvements in performance scores (roughly in the range of 6% to 14%) when the K-L conversion was used. Although the database was relatively small, and therefore care must be taken in extrapolating these accuracy values to larger sets, they provide a strong indication that the color conversion process can have a sizable impact on face recognition performance.

Because the face images had a noticeable increase in contrast as a result of the K-L derived RGB to monochrome transformation, there was a concern that the K-L derived method was doing no more than could be obtained from a common histogram equalization on the color image. To explore this idea, the eigenface performance was also measured with and without the use of a histogram equalization pre-processing step. Each color plane in the original RGB space was enhanced using a standard 256-to-64-bin histogram flattening procedure. The results show that, rather than a performance increase similar to that obtained from the optimized color conversion, the histogram equalization actually produced a severe decrease in accuracy. It is believed that this is due to the global nature of the process, which may have
resulted in a suppression of the facial features that are useful for recognition. We conclude that color histogram equalization is not a useful pre-processing step for eigenface face recognition, regardless of the choice of method for color transformation.

<table>
<thead>
<tr>
<th></th>
<th>Equal-weight RGB</th>
<th>K-L RGB</th>
<th>Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>( R_2 ) – probe in top 2?</td>
<td>0.343</td>
<td>0.371</td>
<td>8.3%</td>
</tr>
<tr>
<td>( R_5 ) – probe in top 5?</td>
<td>0.514</td>
<td>0.600</td>
<td>13.8%</td>
</tr>
<tr>
<td>( R_{10} ) – probe in top 10?</td>
<td>0.686</td>
<td>0.728</td>
<td>6.2%</td>
</tr>
</tbody>
</table>

### 5.6 Conclusions

Most existing face recognition systems operate using monochromatic information alone, even when color information is available. In such cases, a simple and suboptimal conversion process is typically used. Recognition accuracies can be improved if the color-conversion process is selected based on the expected color distributions. We explored three such approaches to determine an improved mapping empirically: Karhunen-Loève analysis of the color pixel distributions, a least-squared-error line fit in RGB space, and a genetic algorithm.

The color-conversion method presented in this chapter is independent of the actual face-recognition approach that is used. For testing purposes, however, we have used the well-known eigenface method. Our experiments using the eigenface method for recognition resulted in performance improvements in the range of approximately 6% to 14%, for a database of 280 color images. Relative distance measurements of Gabor jets of the face area also showed an increase in discriminability of 4% to 6%. Evaluation of the cumulative eigenvalues produced by an eigenface analysis of intensity images and images converted to grayscale form using the computed conversion vector showed a modest, yet consistent improvement in the potential accuracy in retaining only the most important \( n \) basis vectors.
Chapter 6  Quaternion Elastic Graph Matching

6.1 Introduction
To test the validity of the thesis statement that use of color information can improve face recognition accuracy, it is necessary to explore the performance impact of extending a well-known face recognition algorithm. An approach that is popular in the academic literature and in the commercial arena is the elastic graph matching method. In this chapter, the extension of this method to produce the quaternion elastic graph method is described.

The elastic graph method was chosen for several reasons. It is based on applications of the Gabor filter which has some degree of robustness in the presence of image artifacts and degradations. Published reports indicate that it is one of the more accurate methods. Published work already demonstrates an improvement in accuracy by including color in an eigenface implementation (see [Torre99]; a more significant contribution would be achieved by showing the advantage to the use of color with another algorithmic approach. In this chapter, the elastic graph method is briefly described. A key element of this method is the use of facial landmarks, which are usually located manually; section 6.3 describes a novel method of locating these landmarks using an information-theoretic analysis of a face database. The remainder of the chapter details the specific method implemented in the test application: construction of a face model, enrolling a new face, and matching a candidate face image.

6.2 The Elastic Graph Method
A complete implementation of the elastic graph method for face recognition was written. The software implementation allowed for the side-by-side comparison of the use of complex jets from monochromatic images with quaternion jets from color images. Other than the jet extraction and some specifics of the matching, all other portions of the enrollment and matching processes were identical for the color and the monochromatic cases.

The method implemented was a basic elastic graph matching algorithm as described in [Wisko99], with several important differences. Wiskott’s paper described a manual process
for locating the node locations on the face. One of the thesis statements of this research was that a statistical method of selecting the node or lattice points could result in better discrimination. A concept and method were developed, and a separate software application written to accomplish this selection of points. The method is described in section 75.

In the elastic bunch graph method for face recognition, new faces are compared to existing models and, if a node is “significantly” different from those already represented, a new jet model is defined. However, if the new node is “sufficiently” close to one of the existing jets, then it is instead recorded as a new instance of that type of jet. The database of jets that represent the different types of jets seen at each node is called a bunch graph. At each node, the collection of all types of jets seen is called a bunch. Each type of jet in a bunch corresponds to a certain type or appearance of the face feature at that point (similar eyes will result in a similar jet, and so on). This way of representing models depended on a number of heuristic parameters that were difficult to find suitable values for, and that were very likely to require different values for the monochrome and color cases. Therefore, it was decided to forgo the use of bunch graphs so that the two implementations were as close as possible. This departure from the published method is appropriate since the bunch graph adaptation was originally included to increase processing efficiency.

While the original method was only described in terms of Euclidean distance for matching, the current implementation allowed for either Euclidean or Mahalanobis distance. The Mahalanobis distances were computed from the covariance matrix of the jets at all nodes, to normalize each node to the overall variability of the separate jet components at all nodes.

### 6.3 Selection of Lattice points

As modified in [Wisko99], the elastic graph method is based on extraction of features at predetermined locations on the face. The locations are determined manually, and the exact criteria for selection are not fully described. Thus, there is no procedure to follow to ensure consistent results and no indication that the points chosen are in any way optimal.

For the elastic graph matching implementation used in this research, a new approach was taken. In the usual case, there is the potential to acquire more than one image of each person
enrolled (added to the database); this is commonly done in modern biometrics systems to test the similarity of the enrollment images and to detect anomalies. This provides an opportunity to perform some statistical analysis on the population of images. If the images are co-registered spatially, we can develop statistical properties of various regions of the face. From this observation, it was postulated that a statistical analysis might identify candidate face landmarks in a more repeatable manner than is often done.

It should be mentioned that the well-known eigenface method does a form of this type of analysis, as does local feature analysis. However, the eigenface method (and, presumably, local feature analysis) make use primarily of the degrees of distinctiveness between images of different individuals. Of course, the more general question of identification of regions that are well-suited for discrimination should also take into account the degree to which irrelevant image variations are seen.

To examine the problem in a more basic fashion, we state several assumptions.

6. The images in the set to be processed are registered as described in section 7.5.
7. There are multiple images of each of multiple individuals. Generally, there is no restriction on the number of images available for each individual, but the following work has assumed more than two.
8. The illumination and color balance characteristics of the images in the set are approximately consistent.
9. The points at which we want to extract features are those where the image data is, in some sense, more useful for discrimination and less vulnerable to variation among images of the same individual.

This last assumption contains the definition of what is herein called relevance. The relevance of a particular image region is an expression of how useful the information in that area will be in distinguishing the individuals whose images were considered in the computation. We define the relevance as the difference between the “distinctiveness” of images of different individuals and the “similarity” of image sets of different and the same individuals. The terms “distinctiveness” and “similarity” will be replaced by more formal definitions below.
This can be interpreted as follows. If an image region is very similar in images of different individuals, then it is not a useful area to discriminate among them. If another region is very different for images of different individuals, but is also quite different for different images of the same individual, then it is also not a useful area, since the differences observed do not correlate to different identities.

If, however, a portion of the image shows high degrees of difference between individuals, and low correlation between the collections of images of the same and of different individuals, then that portion of the image is highly relevant for discrimination. Let \( \text{class} \) denote the collection of images of the same individual; then \( \text{cross-class} \) relates to images of different individuals and \( \text{in-class} \) to images within the same class. We can then define our relevance metric as follows:

\[
\text{Relevance} = \text{Distinctiveness}_{\text{cross-class}} - \text{Similarity}_{\text{cross-class/in-class}}
\]  

(6-1)

The specific definitions of cross-class distinctiveness and similarity between cross-class and in-class populations remain to be stated.

### 6.3.1 Theoretical Justification

Consider a collection of \( M \) faces, with \( N \) three-color images of each. All images contain \( R \) rows and \( C \) columns. It is possible to define a five-dimensional array of values to completely represent this data:

\[
I(m, n, x, y, p) \text{ where:}
\]

\[
m \in \{1 \cdots M\} \text{ indicates the particular face (the individual)};
\]

\[
n \in \{1 \cdots N\} \text{ indicates the image of that face (the image) – note we make the simplifying assumption that we have the same number of images of each face};
\]

\[
y \in \{1 \cdots Y\} \text{ indicates the y coordinate or row in the image (the row)};
\]

\[
x \in \{1 \cdots X\} \text{ indicates the x coordinate or column in the image (the column)};
\]
\( p \in \{1 \cdots P\} \) indicates the color component (the plane) – note that the planes may be RGB, HSI or any other color representation, and that \( P \) is not necessarily 3.

For convenience, this discussion will assume that there are three planes, \( r, g \) and \( b \), but this is solely for notational convenience and there is no loss of generality. The computation of the relevance image can be performed using images with any number of planes – including 1 (monochromatic images).

We will consider the trace of the covariance matrix (the total variation, [Mardi79]) of a collection of values as the expression of the variance of the set. For notational convenience, \( x \) and \( y \) are suppressed and \( i_{mnp} = I(m,n,x,y,p) \). So, the variance \( V_n \) image of the collection of images of the \( n^{th} \) face is given by:

\[
V_n = \frac{1}{M} \sum_{m=1}^{M} i^2_{n,m} - \frac{1}{M^2} \left( \sum_{m=1}^{M} i_{n,m} \right)^2 + \frac{1}{M} \sum_{m=1}^{M} i^2_{n,g} - \frac{1}{M^2} \left( \sum_{m=1}^{M} i_{n,g} \right)^2 + \frac{1}{M} \sum_{m=1}^{M} i^2_{n,b} - \frac{1}{M^2} \left( \sum_{m=1}^{M} i_{n,b} \right)^2
\]

\[= \frac{1}{M} \sum_{m=1}^{M} \left(i^2_{n,r} + i^2_{n,g} + i^2_{n,b}\right) - \frac{1}{M^2} \left( \sum_{m=1}^{M} \left(i^2_{n,r} + i^2_{n,g} + i^2_{n,b}\right) \right)^2 \]  
(6-2)

We want to characterize the amount of information contained at a given pixel location in this collection of images. The information theoretic concept of mutual information defines the mutual information between two random variables \( X \) and \( Y \) whose probability mass functions are known as \( p(x) \) and \( p(y) \) as:

\[
MI = \frac{1}{MN} \sum_{m} \sum_{n} p_{mn} \log_2 \left( \frac{p_{mn}}{p_m p_n} \right)
\]
(6-3)

We consider the collection of pixel values as an estimate of the probability mass function, as is frequently done. This is appropriate since the cross-correlation of two different collections of pixel intensity values will produce an estimate of the underlying probability correlation, or joint probability.
To evaluate the relative usefulness, or relevance, of the pixel locations in the image for discriminating among faces, we will apply the concept of mutual information. Mutual information is considered to be “a measure of the amount of information that one random variable contains about another random variable. It is the reduction in the uncertainty of one random variable due to the knowledge of the other.” [Cover91]. For a particular location in the image to be of great significance for distinguishing faces of different people, the information that it conveys must be large, or its mutual information with the distribution of the pixel values for all images must be small.

To see this, consider the histogram of pixel intensity values in a particular color plane as an approximation (or estimate) of the probability density function for that pixel location across a set of images [Gonza02]. The distribution of values over two types of images estimates the joint distribution. So, if we are dealing with $p_{mn}$ then $M$ corresponds to the set of all individuals and $N$ to the set of all samples of a particular individual.

Expressing this more accurately, $p_{mn}$ is the probability density function of pixel values across all individuals and all instances, $p_{m}$ is the p.d.f. of pixel values of for a given individual for all instances, and $p_{n}$ is the p.d.f. of pixel values for all individuals.

In this context, what does “mutual information” signify? High mutual information means that the distribution of pixels for all individuals will tell a lot about the distribution for different samples of a particular individual. On the other hand, low mutual information means that the distribution of pixel intensity values for different individuals has little to do with the distribution for a given individual.

The best pixels for discriminating faces of different individuals will be those where the cross-individual variance is high and the mutual information between the cross- and intra-person distributions is low. Therefore, we form the following quantity:

$$V_n - MI = \frac{1}{M} \sum_{m=1}^{M} \left( i_{mn}^2 + i_{mng}^2 + i_{mnb}^2 \right) - \frac{1}{M^2} \left( \sum_{m=1}^{M} i_{mn} + i_{mng} + i_{mnb} \right)^2$$

(6-4)
Chapter 6 – Quaternion Elastic Graph Matching

6.3.2 Implementation of the Analysis

The proposed algorithm has been implemented in C++, using the Microsoft Vision SDK with matrix support. The images in the database are used to compose a set of matrices storing statistical information for each pixel location in the image:

- the sum of pixel values for one image for each subject (used to compute the mean image);
- the sum of the squares of pixel values for one image for each subject (to compute the variance);
- the cross-sum, the sum of pixel values across all subjects for each location in the image, used to approximate the partial probability $P_n$ (note that there are the same number of independent cross-sum matrices as there are images of each subject);
- the inter-sum, the sum of pixel values for each image of a single subject, used to estimate the partial probability $P_m$ (note that there is one inter-sum matrix for each subject in the database).

These “raw” statistics matrices are used to compute two final matrices, the gray-level (or color component) variance and the mutual information, which are combined to form the result, called the relevance image. Brighter areas in this image have high cross-subject variance but low mutual information between the inter-subject distributions and the cross-subject distributions. This analysis was performed on several databases using both color and monochrome pixel values; results were very similar for databases of sufficient size (larger than 32 subjects with at least 8 images of each subject). An example relevance image can be seen in Figure 16.
Several conclusions can be drawn from this analysis. Most importantly, the areas of the face image that will be most useful in discrimination can be identified by inspection, as bright regions in this result image. Some of the key points would be chosen by intuition: the centers and corners of the eye region, the mouth and the point of the chin. However, other key regions are apparent from this analysis (that would not have been as apparent to me): the area between the eyes, the eyebrows, the corners of the nose and the area of the hairline.

A second effect that is readily observed is the variation caused by the presence of glasses and differences in hairline. In the databases considered, each individual that wore glasses was imaged only with glasses on; there was no attempt to take some images with glasses and some without. The hairline was also reasonably consistent from image to image for the same subject. Therefore, these features (presence or absence of glasses, height of forehead hairline) were detected as good differentiation areas, since they are very consistent from sample to
sample (high cross-person variance with low mutual information between the cross-person and intra-person probabilities).

Interestingly, face features are less “relevant” in the areas that receive the most illumination. While this may be partially due to increased contrast in the dimmer areas of the face, where the intensities are not limited by the dynamic range of the imager, on this database the cause is much simpler (and less desirable). The level of illumination on the right side of the faces is lower for a number of individuals in the database; therefore, this appears as a discriminating element. Clearly this is not a useful feature to extract in the general case.

Finally, feature regions that are closer to the center of the face are better defined than areas near the perimeter of the face. This is due to the limitations on the face registration (note especially the clear definition of the result images around the eye landmarks where the positional alignment is the best). For this same reason, face landmarks that are far from the center of the image should be avoided. Features near the edge of the face are problematic in other ways: unpredictable obstruction from hair, greater variations in apparent illumination because of the curvature of the face, and (for some imaging configurations) inferior image detail due to optical defocus.

### 6.4 Constructing a Model

Using the results of the statistical lattice selection, we construct the face model geometric form. Twenty-four points were chosen from the relevance analysis, based on the variance / mutual information result. Initially an automatic process was defined and implemented to identify the specific points to compose the face model’s graph. However, this approach was discarded in favor of a semi-automated process for defining the face model. There were several reasons for not continuing with a fully automated process for defining the lattice:

The procedure for locating the most significant peaks in the relevance image required several heuristic factors that strongly influenced its performance: thresholds for peak detection, the minimum distance between peaks, the method and weight of “derating” the perimeter region of the face and the number of vertices to extract.
Even if the vertices are extracted automatically, it is still necessary to determine which vertex pairs to connect with a graph “edge” – that is, a feature-to-feature distance that will be significant in the edge model graph matching.

As observed earlier, some apparent relevance factors are, in fact, artifacts of the process – for example, the illumination level of the right side of the face (for this database). By completing the model construction process manually, we can intelligently ignore artifacts of this type.

So, given the results of the relevance analysis on the database, we construct a face model by identifying a number of locations in the image for jet locations. The face model is built on the underlying framework of an undirected graph, with the possibility of several edges for each vertex. The first step is to locate the vertices. By inspection of the relevance images for several databases, the following 24 points are chosen.

- Center of the hairline
- Left corner of the left eyebrow
- Center of the left eyebrow
- Right corner of the left eyebrow
- Left corner of the right eyebrow
- Center of the right eyebrow
- Right corner of the right eyebrow
- Left corner of the left eye
- Center of the left eye
- Right corner of the left eye
- Point between the eye centers
- Left corner of the right eye
- Center of the right eye
- Right corner of the right eye
- Left edge of the base of the nose
- Point of the nose
- Center of the base of the nose
- Right edge of the base of the nose
- Left corner of the mouth
- Center of the mouth
- Right corner of the mouth
- Jawline directly below the left eye
- Center of the jawline
- Jawline directly below the right eye
Once the vertices to be used in the face model are identified, we identify a number of pairs of vertices to be connected with edges in the face graph. Vertices are connected if the distance between them is a significant feature to use in extraction. Note that only the spatial distance is used in recognition, rather than the relative orientation (only the length of the offset vector is considered, not the direction). This provides some rotational invariance to the face model, up to about 15 degrees where the response of the jets themselves will start to change significantly.

To choose the vertices to connect in the face graph, we select inter-feature dimensions that are not subject to great variance for the same subject but may differ from subject to subject. Several inter-feature distances are easily seen as good discriminators, by analogy to human recognition of faces: width of the mouth, distance between the eyes, distance from the eyes to the hairline, distance from the mouth to the jawline. Other vertices are selected either by additional inspection of images (width of the nose and of the eyebrows) or because the features are relatively close together and therefore should be relatively easy to locate precisely (distance from eyes to eyebrows).

The final face model is defined to have 24 vertices and 26 connecting edges. A representation of the face model geometry is shown in Figure 17, where the locations of the vertices and edges are overlaid on a sample image.
6.5 Enrolling a New Face

The face model for a particular face consists of a number of descriptors. Some are common to all modes in a face database:

- The type of the model – Complex (monochrome) or Quaternion (color);
- The number of landmarks in the model;
- The nominal location of each landmark in the "default" image;
- The number of edges or connecting segments between landmarks;
- The pairs of landmarks connected by each edge.
Other descriptors are specific to each face in the database, and contain the features used for recognition:

- The jet (either complex or quaternion) extracted from the teach image at each landmark location;
- The actual spatial distance along each edge in the model;
- To add a new subject to the population used for matching, the following steps are taken:
- Common attributes such as type and number of landmarks are determined from the database properties;

At each nominal location, a search is conducted in a small neighborhood to find the location at which the jet is most similar to the "prototype" jet at that landmark. The jet data is stored for that landmark. For each edge, the actual spatial distance is stored. The jet data is added to the face database’s jet covariance matrix for use later in computing Mahalanobis distances.

### 6.6 Matching a New Face

The process of matching (or attempting to match) a candidate face against one or more faces in the population is done by computing a distance measure between the features from the candidate face and features contained in the face models in the database.

#### 6.6.1 Verification and Identification matching

Biometrics systems often make a distinction between the operation of a verification system (where a presumed identity is known and the candidate is being matched against only one model) and an identification system (where the candidate is matched against all models in the database to see if any match is sufficiently close to declare a successful identification). Of course, no identification is possible unless the individual contained in the candidate image has already been enrolled in the system, and at least one model in the database represents this individual.

Both forms of matching involve a *threshold*, which is some measure of the closeness of match required between a candidate and a face model for the system to report a successful match. For example, in the case of an identification match, it is quite likely that the face model in the
database which matches most closely with the candidate image is not sufficiently close to the candidate (presumably, this would mean that the person in the candidate image is not in the database). The threshold should be set to detect most false matches of this type, without rejecting any correct matches. In systems where the final match determination is a distance measure in some feature space, the threshold is a maximum distance allowed between the candidate feature vector and any model's feature vector before a match would be reported. It is possible for more than one model to match a candidate image to less than the match threshold. In this case, the closest match would be reported.

The implementation used in this research did not use any threshold. In all cases, the closest match was reported and used in the testing. In Chapter 8 on performance, results for Receiver Operating Characteristic curves are reported. For a distance-based system, this is a reflection of the system's operation for various values of threshold. These results were obtained by reporting out the system's match scores in terms of the distances. By processing these results, it is possible to derive the corresponding error rates for any threshold value.

A simple way of implementing an identification system is by repeated verification matches, once to each model in the database. If any matches meet the threshold, then the closest is reported as the correct identification. More sophisticated systems, particularly those supporting large databases, may use a preprocessing step to limit the members of the database against which feature-based matching is actually performed. This technique is called database filtering, and it is primarily used to increase the system's throughput (the effective match rate) at the cost of a very small increase in false-non-match rate. No form of filtering is used in this implementation.

### 6.6.2 Distance measure used in matching

The implementation used in this research performed matching using the distance from the candidate's feature vector to the presumed identity's feature vector. Both Euclidean and Mahalanobis distance were used, for different trials. The feature vector used was composed of the jets extracted at each node and the spatial distances along each edge included in the model. Since these were two different forms of data, with different ranges, two scale factors $\alpha$ and $\beta$ were included to normalize the two types of feature. Generally, $\alpha$ was set to 1, so that changes
to $\beta$ would put more or less importance on the matching of the spatial distances between selected face landmarks.

The distance between the feature vectors for a candidate image $v$ and the $w^{th}$ model is computed as follows, where $J_{vn}$ is the $n^{th}$ jet for $v$ and $d_{ve}$ is the distance along the $e^{th}$ graph edge for $v$.

$$D_{v,w} = \alpha \cdot \frac{1}{N} \sum_{n=1}^{N} \| (J_{vn} - J_{wn}) \| + \beta \cdot \frac{1}{E} \sum_{e=1}^{E} |d_{ve} - d_{we}|$$

(6-5)

### 6.7 Conclusions

A version of the elastic graph face matching method has been implemented to evaluate the proposed use of color. This method is based on the extraction and matching of Gabor feature vectors called jets from specified locations on the face image. The locations for these feature extraction operations form nodes in a graph. Both the Gabor filter responses at each node, and the distances between certain nodes in the graph, are features used for recognition.

A new procedure for location of face landmarks has been introduced. This method takes advantage of the fact that there are several images of each individual available. By computation of the image pixel variance, we find areas of great difference from subject to subject. By computation of the mutual information between intra- and cross-class pixel distributions across the collection, we identify areas where several images of the same subject have different statistical properties than images of other subjects. A combination of these two types of quantities gives what is here called a relevance image. Areas of high intensity in this image are good locations for facial landmarks.

These landmarks are used to construct a face model. New subjects are added to the database by constructing a face model. This is done by collecting the responses of a family of Gabor filters at each face model and concatenating them with some spatial distance information. An active procedure offsets each node slightly to find the location in the candidate image that most closely resembles the face feature identified with that node.
Matching is performed using a distance measure calculated from the jet features and the spatial distances, weighted so as to equalize the sensitivity of the process to jet and distance features. Verification or 1-to-1 matching is done by comparing the subject and the asserted identities' feature models. Identification or 1-to-many matching is here implemented as a simple series of 1-to-1 matches.
Chapter 7  Testing Methodology and Implementation

7.1  Introduction

The thesis statements presented in section 1.6 must be substantiated experimentally:

- the use of color information can improve face recognition accuracy;
- extending the Gabor filter to color images can increase accuracy;
- statistical methods can improve face landmark location.

To demonstrate the validity of these three statements, extensive performance testing of the two elastic graph matching implementations has been conducted. The assertion is that if the quaternionic elastic graph matching algorithm, using quaternionic Gabor filters operating on color images, can demonstrate accuracy that is significantly better than achieved by the monochrome elastic graph matching algorithm, then the three thesis statements are supported.

This chapter presents the testing strategy and methods used. The testing strategy is discussed in section 7.2, while section 7.3 provides some information on the software implementation itself. One issue associated with research into the use of color for face recognition is the relative paucity of suitable databases of color face images; section 7.4 describes the databases that were obtained and used in this work. Section 7.5 discusses the face registration task, while section presents the conclusions of this chapter.

7.2  Testing Strategy

To experimentally verify the three main theses, a software face recognition testbed was implemented. As mentioned earlier, the assertion is that the three theses will be supported if the proposed color recognition algorithms achieve accuracy that is generally superior to that of the monochrome implementation. To draw this conclusion it is imperative that the two implementations vary only in the features extracted from the images; there must be no differences in the subsequent analysis, modeling and matching.

The goals of this research effort are to evaluate the use of color, and color Gabor filters, for face recognition. Therefore, not all components of a face recognition system were included.
Most notably, automated face registration was not implemented. Most often, this involves the location of the eye centers and registration. However, to use the face databases available, it was necessary to register the images. As a preliminary step to the research here described, the images were manually registered by locating the eye centers and registering the faces as described in section 7.5.

It should be noted that the monochrome implementation used the optimal color axis projection as described in Chapter 5 to transform the color images in the face databases to monochrome images. Therefore, even the monochrome implementation makes more use of the color information in the face images than is commonly done today. The test efforts described below investigated the benefit of processing all of the color information, as compared to using an optimally-converted color face image database.

7.3 Software Implementation

To allow the greatest flexibility and ease of development, C++ was selected as the environment for the software implementation. Other options were considered, such as MATLAB and Java. From the beginning of the work, the author anticipated that the software required would be significant, and this has proven to be true (the three major applications total to over 8500 lines of C++, written as part of this effort). MATLAB does not enforce nor allow enough of a structured programming style to facilitate such a large development. Java is awkward for use in applications that deal at the pixel and direct data level. Therefore C++ was selected as the language for the implementation.

The Microsoft Vision SDK [Micro00] provides a framework for the various software applications used in this research. It defines convenient image and pixel objects, simple image input and output functions, and even supports matrix operations including inversion and eigenvalue / eigenvector extraction. The Vision SDK is only supported on Visual C++ version 6.0.

Three major applications have been written: “Lattice” is used to process sets of images and compute the relevance information described in section 6.1; “Enroll” is used to build a face model; and “Locate” performed the vertex location and matching, and was later extended to
run the complete database matching tests. A fourth software application called "ROCtest" was written to expedite the reporting of match distances (match scores) for generation of receiver operating characteristic curves. Very little effort was expended in designing and implementing clean user interfaces for these applications; they are all simple Single Document Interface or Multiple Document Interface applications with characteristic Windows look and feel.

The functional portions of the code were carefully designed using an object-oriented approach. To give some idea of the capabilities of the underlying objects and their interfaces, a UML class diagram is contained in Appendix D.

The major goal of the software implementation was to provide a side-by-side test of monochrome and color elastic graph face recognition. This was achieved by defining an abstract Jet base class, from which we derive two tangible classes for CJet (complex jets, for application to monochrome images) and QJet (quaternion jets to be used on color images). The methods for enrollment, matching and face model manipulation make extensive use of polymorphism to minimize the areas of software variation.

### 7.4 Face Databases

One of the challenges in the research of color face recognition has been the lack of available databases in color. Often, those databases that are available are of poor quality, with over-compression of the images or wide variations in the illumination. Recently, several good quality databases have become available. The five color databases that have been used at various points in the current research are listed in the following table.

The three databases that were the most useful in the testing of the face recognition implementation are described briefly below. The CVL and Essex databases are only available in JPEG compressed form. The performance results show similar gains in performance for compressed and uncompressed images, though this question should be the subject of further study. In particular, the parameters used for the image compression of color images should be carefully considered.
Table 6 - Summary of face databases used in this research

<table>
<thead>
<tr>
<th>Database</th>
<th>CVL (Ljubljana)</th>
<th>Univ of Essex</th>
<th>PIE (CMU)</th>
<th>AR (Purdue)</th>
<th>Univ of Oulu</th>
</tr>
</thead>
<tbody>
<tr>
<td># of images</td>
<td>798 (114 used)</td>
<td>7900</td>
<td>41368</td>
<td>&gt;4000</td>
<td>2112</td>
</tr>
<tr>
<td># of individuals</td>
<td>114</td>
<td>395</td>
<td>68</td>
<td>126</td>
<td>111</td>
</tr>
<tr>
<td>format</td>
<td>24-bit color .jpg</td>
<td>24-bit color .jpg</td>
<td>24-bit RGB .ppm</td>
<td>24-bit RGB raw .ppm</td>
<td>24-bit RGB .bmp</td>
</tr>
<tr>
<td>image size</td>
<td>640W x 480H</td>
<td>180W x 200H, 196W x 196H</td>
<td>640W x 486H</td>
<td>768W x 576H</td>
<td>428W x 569H</td>
</tr>
<tr>
<td>pose variation?</td>
<td>7 different</td>
<td>some</td>
<td>13 different</td>
<td>some</td>
<td>minor</td>
</tr>
<tr>
<td>illumination variation?</td>
<td>no</td>
<td>minor</td>
<td>43 different</td>
<td>some</td>
<td>16 different</td>
</tr>
<tr>
<td>expression variation?</td>
<td>some</td>
<td>minor</td>
<td>4 different</td>
<td>some</td>
<td>with/without glasses</td>
</tr>
</tbody>
</table>

7.4.1 The Ljubljana CVL database

Dr Peter Paar and others at the University of Ljubljana (Slovenia) have provided a database of 114 individuals, containing 7 images of each ([Ljubl02]). The illumination is quite uniform, but there are a variety of viewing angles for each subject from left side through frontal view to right side. For the purposes of my work, only the frontal views are used. The images are 640 by 480 color in JPEG format.

7.4.2 The University of Essex database

The University of Essex (UK) face databases ([Space96]) contain 20 images each of 395 individuals. The subjects represent a wide distribution of ethnic groups and both genders, but nearly all subjects are between 18 and 20 years old. The images in the Essex database are JPEG-compressed 24-bit color images.
7.4.3 The CMU PIE database

Carnegie Mellon has compiled a database of face images under a variety of pose, illumination and expression conditions ([Sim01]). The database contains 24-bit RGB images of 68 individuals, each under 13 different poses, 43 different illumination conditions and with 4 different expressions. The pose images were taken (virtually) simultaneously from 13 different cameras to ensure that the data represented only variations in pose, and not other factors. The image format is raw raster scan; each image is 768 by 576 pixels.

7.5 Face Image Registration

Before use, the images are converted to an easily manipulated format such as JPEG or BMP. The face images are then registered in accordance with emerging standards for face image transmission (see [Griff02]). The centers of the eye sockets (not necessarily the centers of the pupils) is placed at image pixel locations (91, 145) and (150, 145), with the origin of the coordinate system at the upper left. A typical canonical image is shown in Figure 18.

Figure 18 - Canonical Face Image as described in [Griff02]

To produce this image from a general frontal face image, several steps are taken.
The image is padded with a margin of black pixels;

The eye positions are determined, either automatically or manually;

The image is rotated to horizontally align the eyes;

The image is scaled so that there are 60 pixels between eye centers;

The image is translated and cropped to an overall size of 320 high by 240 wide, with the eye centers at (91, 145) and (150, 145).

This procedure is shown graphically in Figure 19 (from [Griff02]).

![Figure 19 - Image Rotation, Scaling and Cropping (from [Griff02])](image)

A set of MATLAB m-files was used for user interface, image transformations and file I/O for the manual registration procedure. All test images used in this work are preprocessed in this manner; all eye locations have been entered manually. This eliminates possible variations due to inaccuracies in the automatic location of the eye centers; personal interviews with commercial developers have identified automatic eye location as one of the major causes of error in deployed commercial systems.

### 7.6 Conclusions

A complete face recognition implementation has been developed to provide a test environment for the thesis statements. The only function not implemented that is usually present in a commercial face recognition system is automated eye location and face
registration. This was omitted as a potential source of error, and as unnecessary to support off-line performance testing.

The implementation allows for side-by-side testing of the monochromatic and color recognition methods described earlier. The recognition applications are written in C++, and make use of the Microsoft Vision SDK and the Microsoft Foundation Classes. Face registration was performed off-line using software written in MATLAB.
Chapter 8  Performance Comparison

8.1  Introduction

The face recognition system has been implemented to support both monochrome operation using complex Gabor filters and color images using quaternion Gabor filters. Extensive effort was devoted to ensuring that the two methods differ only in the actual filters used to extract the jet information. To evaluate the relative benefit of the quaternion jet operations, we need only run the database construction and matching algorithms on a number of bases and present the results.

In this chapter, the results of two types of testing are presented. Section 8.2 describes the results of measuring the verification, or one-to-one matching, accuracy of the two implementations. In conformance with usual biometric system testing, this comparison is done on the basis of the receiver operating characteristic (ROC) curves. Section 8.3 presents the identification or one-to-many matching test performance. This is described in terms of the match rate achieved for several databases. Finally, the last section of the chapter discusses the meaning of the performance comparisons and draws some conclusions.

8.2  Comparison of Verification Performance

Evaluation of the accuracy of any biometric system cannot be based on any single metric. Since the problem is fundamentally a pattern recognition task, there are inevitably one or more decision thresholds. Comparison of a biometric sample to its presumed match results in the question “is the match close enough?”

By adjustment of the thresholds, the performance characteristics of the system can be tuned to the application requirements. In some situations, it is most important to prevent any incorrect match; consider an access control system on a high-security area, for example. The probability of a false match must be reduced as much as possible, even at the expense of an increase in the probability of a false non-match. In the access-control situation, it is acceptable to require valid users to re-present their biometric frequently, if by doing so we reduce the chance of allowing access to an intruder. The probability of a false match is estimated by the observed False Match Rate or False Accept Rate (abbreviated FAR). For an experiment of $N$
attempted false matches, the $FAR$ is computed as the ratio of incorrect matches to the total number of trials, $FAR = \frac{n_{accept}}{N}$, and is often expressed in percent. A False Accept Error is sometimes referred to as a Type II error (from statistical analysis), considering that the statistical hypothesis is a match between the biometric sample and the presumed identity. In the case of a False Accept Error, the hypothesis of a match is indeed False (there is no match), but the system incorrectly accepts the hypothesis.

However, the opposite error is also important. A Type I error, where the hypothesis of a match is true but the system does not accept the match, is commonly referred to as a False Reject Error. The False Reject Rate ($FRR$) for an experiment of $N$ attempted correct matches is then $FRR = \frac{n_{reject}}{N}$. The $FRR$ may be of importance where we want the system to incorrectly reject as few authorized users as possible. For example, Walt Disney World uses biometric identification (based on hand geometry) to verify users of annual passes to the park, as a way of preventing sharing of passes among non-payees. Since these passes are purchased by their best customers, it is important that the system not inconvenience authorized users, even at the cost of an increase in the false accept rate. It is clearly better for only one customer in a hundred to re-present their hand, or seek assistance, though the probability of entry by a person using someone else’s pass is as high as 0.1. This probability is still low enough to discourage fraud.

A biometric system can be tuned to the preferred level of performance, usually expressed in $FAR$ and $FRR$, by adjusting one or more thresholds. There is a direct trade-off between these two types of errors. If the thresholds are set to require nearly an exact match, then the False Accept Rate will be low, but the False Reject Rate will be high. Conversely, more “relaxed” thresholds will increase the $FAR$ but reduce the $FRR$. Every threshold setting will generate a related pair of $FAR$ and $FRR$. Plotting these pairs in the $FRR$-$FAR$ plane results in what is known as a “Receiver Operating Characteristic”, or ROC curve, also borrowed from statistical analysis. This curve may have an unfamiliar appearance compared to ROC curves from other fields, where the curve is commonly high at the left and declining to the right. In biometrics, it is common to plot a modified ROC curve, composed of the total verification rate,
$TVR = \frac{n_{corrVerify}}{N}$, plotted versus FAR. This follows the practice of the FRVT test reports ([Phil03]). An example of an ROC curve is shown in Figure 20. As described in section, the test implementation does not directly support a match threshold which is usually implied by the reporting of an ROC curve. Instead, all test runs reported the distance to the closest $n$ matches in the database in a candidate list. By collecting and sorting this output data, we can generate parametrized ROC curves with the match threshold as the implied or hidden parameter. This post-processing of the raw results was done in a set of Microsoft Excel spreadsheets to allow ease in charting the data.

Even though verification performance is more complex than can be accurately reflected in one metric, reducing the information in an ROC curve to a single number is simply too convenient. The Equal Error Rate ($EER$) is often used as a one-dimensional measure of the accuracy of a verification system. This is the rate at which the $FAR$ and $FRR$ are the same. Generally, no biometric system is operated at this point, but the assumption is made that systems with lower $EER$ will have lower error rates in the useful portion of their ROC curves. The $EER$ shown in Figure 16 is approximately 28%, which is higher than that achieved by most commercial systems.
Figure 21 shows the relationship between the ROC curve for the monochrome and color implementations. Recall that the two implementations are identical in all respects other than the underlying recognition features and the related parameter settings. It can be readily seen that the ROC operating performance of the color-based verification is significantly higher than the monochrome operating characteristic. Specifically, the Equal Error Rate using color features is 14.04%, approximately ½ of the EER of 28% measured for the monochrome process.

The performance increase can be stated in another manner that is probably more relevant to the real-world use of biometric systems. For a given False Accept Rate, the resulting True Verification Rate ($TVR$) is generally higher, and this increase is a directly measurable improvement in the performance of the system. As an example, the test data shows that, for a 1% False Accept Rate (not an unreasonable figure), the monochrome algorithm provides a True Verification Rate of approximately 39%, while the color algorithm achieves a True
Verification Rate of 44%. For a FAR of 5%, the monochrome algorithm will achieve a 50% TVR, while the color algorithm yields a True Verification Rate of 65%.

For comparison, Figure 22 shows the identification performance of the test implementations plotted on top of the verification ROC curves for the commercial systems evaluated in the FRVT 2002 tests ([Phil03]). This plot shows clearly that the color implementation performance is comparable to several commercial products, though it is significantly below that of the top three performers.
Chapter 8 – Performance Comparison

Figure 22 – Measured ROC compared with monochrome FRVT2002 results from [Phill03]

As a means of quickly comparing the verification performance of the two approaches for several face databases, Table 7 shows the Equal Error Rate (EER) for both monochrome and color analysis for the several databases. The color performance is documented for both Euclidean distance and Mahalanobis distance for classification. This was indicated by the wide variation in the average magnitude of the components of the color jets for different scales.

Table 7 - EER comparison - color vs. monochrome

<table>
<thead>
<tr>
<th>Database</th>
<th>DB size</th>
<th>CJet EER Euclidean distance</th>
<th>QJet EER Euclidean distance</th>
<th>QJet EER Mahalanobis distance</th>
<th>Maximum Improvement (in red if &lt;0)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ljubljana</td>
<td>114</td>
<td>28.5%</td>
<td>14.0%</td>
<td>11.1%</td>
<td>17.4%</td>
</tr>
<tr>
<td>Essex 94</td>
<td>132</td>
<td>22.1%</td>
<td>12.9%</td>
<td>12.3%</td>
<td>9.8%</td>
</tr>
<tr>
<td>Essex 95</td>
<td>71</td>
<td>16.5%</td>
<td>14.5%</td>
<td>10.8%</td>
<td>5.7%</td>
</tr>
</tbody>
</table>
8.3 Comparison of Identification Performance

To compare the identification (or one-to-many matching) performance of the Complex (monochrome) and Quaternion (color) forms of the Gabor graph method, identical sets of images were used to evaluate the ROC characteristics. The results are shown in Table 8. In the table, "rank 1" indicates that the correct match was the top of the candidate list returned from the matching, sorted in order of increasing distance. The column titled "top 5" indicates that the correct match was in the top 5 of the candidate list.

<table>
<thead>
<tr>
<th>Database</th>
<th>DB size</th>
<th>1: many Test</th>
<th>CJet Results</th>
<th>QJet Results</th>
<th>Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>rank 1  top 5</td>
<td>rank 1  top 5</td>
<td>rank 1  top 5</td>
</tr>
<tr>
<td>Ljubljana</td>
<td>114</td>
<td>Euclidean distance</td>
<td>36.0% 51.7%</td>
<td>39.5% 69.3%</td>
<td>3.5% 17.6%</td>
</tr>
<tr>
<td>Ljubljana</td>
<td>114</td>
<td>Mahalanobis distance</td>
<td>38.6% 53.5%</td>
<td>56.1% 84.2%</td>
<td>17.5% 30.7%</td>
</tr>
<tr>
<td>Essex 94</td>
<td>132</td>
<td>Mahalanobis distance</td>
<td>41.5% 54.7%</td>
<td>63.3% 80.4%</td>
<td>12.2% 25.7%</td>
</tr>
</tbody>
</table>

The identification performance of the complex implementation is not very high. A successful identification rate of 36% would not be useful in many actual operating scenarios! Clearly, commercial recognition systems using this technology have a number of optimizations and enhancements that are not included in my test implementation. Several possible areas for optimization were identified during this research, but were not implemented because they would have introduced additional areas of difference between the color and monochrome algorithms.

1. This implementation assigned equal weights to all nodes in the model, though simple inspection showed that some landmark locations were more reliable for discrimination than others. This could be easily accommodated by specifying different weights in the distance computation depending on the observed variance and stability of that node's jets. However, inspection showed that the jets did not vary in
the same manner for the color and monochrome implementations, so that each node would have, in general, had different weights in the two implementations. This was seen to be a significant difference that would have weakened the comparison between the two methods.

2. The number of jet locations was limited to 24, while some elastic implementations use as many as 40 nodes. This was done to reduce the overall processing time. Since transform-based methods were not used to speed up the Gabor feature extraction, execution time for a complete test tended to be long (several hours on even these modest databases). This led to a desire to limit the number of nodes.

3. Most significantly, a simple minimum-distance classifier was used in both implementations. In the large number of dimensions involved (80 or 160) it is reasonable to expect that more sophisticated classification methods would be indicated.

It is almost certain that there are other optimizations and enhancements that are routinely used with elastic graph methods that are not included in my implementations. However, we are able to see the significant performance increase associated with the use of color. Notice that the improvement in rank 1 performance ranged from 3.5% to 12.2%, while the percentage of the time that the correct match was in the top 5 increased by 17.6% to 30.7%. The use of color with the Mahalanobis distance resulted in rank 5 performance of up to 84% which is certainly competitive with many published results for face identification.

It is interesting to see the effect of the use of the Mahalanobis distance. The transformation is done using a covariance matrix computed for all nodes, for all images in the database. Thus, it has the effect of equalizing any components with higher range. The improvement obtained by use of the Mahalanobis distance for the complex implementation was small relative to the improvement seen in the quaternion case. This would suggest that the components of the QGF responses were more unbalanced. Inspection showed that this was the case. The components in the QGF responses at higher values of frequency were much higher in range, suggesting that the color images were more noisy. This could be explained by several factors. Computation of the intensity plane involves an averaging process, providing a reduction in noise in the image that would be expected to be on the order of \( \sqrt{3} \). In addition, it is well
known that the blue plane is significantly more noisy for most imagers, due to the lower response of silicon in the blue areas of the visible spectrum. Finally, the very notion of computing rotation angles between color vectors raises the possibility that computation of the QGF responses involves mathematically less stable operations than the usual complex Gabor filters. This may be an indication that further work is indicated.

8.4 Conclusions

The matching performance of the test implementation has been measured for both the complex Gabor filter and the new quaternionic Gabor filter. Improvement in accuracy is observed for both verification (1-to-1) matching and identification (1-to-many) matching. The specific improvements in accuracy vary from test to test, but range from 5% to 17% for verification and from 3% to 30% for identification. Many tests showed improvements in the range of 12% to 17%. Receiver Operating Characteristic curves have been generated for the complex and QGF tests. They show an increase in operating performance, indicated by a higher ROC curve, over most of the operating range.
Chapter 9  Conclusions

9.1  Summary

This research effort has extended the well-known graph matching method, based on Gabor filter feature extraction, to color images. In particular, the three Thesis statements have been demonstrated:

- Use of color information can increase accuracy beyond that achievable using monochromatic information only;
- Extending Gabor techniques to color can increase accuracy;
- Statistical methods can improve face landmark selection.

The key contributions of this research are as follows:

- Selection of Hypercomplex representation for color pixels;
- Formulation of the new Quaternionic Gabor filter;
- Design of a quantitative procedure for finding the relevance image for a face database.

The following work was performed in support of these contributions:

- Implementation of C++ classes for quaternions, QGF and other basic components;
- Implementation of a monochrome face recognition technology;
- Extension of the face recognition software to color images;
- Performance testing and analysis.

9.2  Thesis Statements

Each of the thesis statements presented in Section 1.6 is here examined in view of the findings of this research effort.
9.2.1 Use of Color Information Can Increase Accuracy

The results summarized in Section Chapter 8 show a consistent improvement in accuracy by the use of color features. The improvement in one-to-one (verification) accuracy ranges from a 5% to 17% decrease in the Equal Error rate. In other words, if the system specification were to dictate that the probability of a False Accept Error and a False Reject Error should be equal, then the use of color reduces the error rate from 24% to approximately 6%. As this result implies, the ROC curve for the color Gabor implementation is significantly higher than the corresponding curve for the monochrome implementation; see Figure 21. For a given False Accept Rate, the color recognition algorithms show approximately half the False Reject Rate of the monochrome algorithms.

One of the most commonly stated reasons for only using color information for face recognition is that the potential gains in accuracy are small. Based on the implementation and testing described in this dissertation, this assertion is not true. It is not being concluded here that extension of the Gabor filter is the optimal method of processing color images for face recognition. However, it has been shown that color features can increase accurately of at least the elastic graph method significantly.

The other reasons for lack of use of color for recognition still exist. Compatibility with existing databases and equipment, need to use color cameras, increased processing requirements and greater need to control the spectral characteristics of illumination are significant factors. Yet, given that face recognition seems to be limited in deployment by both real and perceived accuracy limitations, many situations will justify some increased cost and complexity in order to achieve the highest possible level of performance.

The conclusion formed here that color information does increase face recognition accuracy is consistent with the results of [Torre99] and [Jones04].

9.2.2 Extending Gabor Techniques to Color Can Increase Accuracy

The second thesis statements relates to a specific method for implementation of color face recognition. Use of Gabor filter responses at key facial landmark locations is the basis for many published methods for face recognition, as well as several commercial face recognition
products. The idea was postulated here that the method would lend itself to use with color images.

The Gabor filters used for extracting features are complex functions, intended for application to single-valued monochrome images. This research has led to a novel extension of Gabor functions to the four-component hypercomplex domain. The implementation and performance testing have demonstrated that the application of these functions to color images can produce a sufficiently rich set of features for recognition tasks. An important outcome of the work is a specific encoding of color pixels and a selection of Gabor functions that produce a quaternionic response of specific form. In this form, the real part is the vector dot product of the color pixel and the Gabor filter component, while the hypercomplex components are the inverse cross product of the pixel and the Gabor filter.

Thus, the result of the filter application at each point contains information on both the projection of the color pixel into the axis of the Gabor coefficient and the vector angle between the two colors. Similarity between the image color and the fundamental color vector of the Gabor filter can be measured in this manner.

The performance data and analyses confirm that the extension of this Gabor technique provide significant gains in accuracy. Note that the Gabor recognition implementation is not comparable in performance to the best published results for Gabor-based methods, or to the leading commercially available systems. There are two likely reasons for this: the emphasis in this implementation was on creating a workable “testbed” for comparison of the performance potential of monochrome and color features, and most commercial systems involve proprietary optimizations and enhancements that are not published.

9.2.3 Statistical Methods Can Improve Face Landmark Location

The third thesis statement is that the usual practice of choosing face landmarks by inspection is sub-optimal. Most Gabor implementations that are well-described in the literature rely on jet locations chosen manually, based on experience and common sense. Since face recognition databases contain many images, the potential for some statistical analysis was obvious.
This research has described a method for choosing the most relevant areas of a face image by statistical analysis of the pixel values of a large number of registered images. The criterion for selecting an image region as “relevant” is that the area should have both high image variance among images of different subjects, and low mutual information between the pixel distributions of images of the same subject and images of different subjects. These pixel areas then have high variability and low relation among different subjects.

This technique was used to generate a “relevance” image, where the intensity values were an indication of the relevance for discrimination, as computed by high variance and low mutual information. Some attempts were made to process this relevance image completely automatically, but this was not productive due to the large number of heuristics involved in the analysis. Alternatively, the relevance image was examined manually and facial landmarks placed at the high intensity areas of the relevance image. While some of these landmarks were at or very near to the points that are “traditionally” chosen for face analysis, several others that were indicated by the relevance image are new, and not obvious. The resulting landmarks form the basis for the face models used in this research effort.

9.3 Future Work

It has been demonstrated that quaternionic Gabor filters, operating on color images, produce an increase in face recognition accuracy of from 3% to 17% on the image databases considered. There are several areas of future work that have the potential for significant results.

Face landmark locations were identified by identifying areas of the face with high inter-class variance and low mutual information between the inter-class and intra-class pixel distributions. This method revealed several unique areas of relevance on a face image. The more general idea of using a combination of image pixel variance and elements of information theory to identify locations in the image for feature extraction has promise for future work. Recall that the conceptual basis for the relevance image calculation was:

\[ \text{Relevance} = \text{Discrimination}_{\text{cross-class}} - \text{Similarity}_{\text{in-class/cross-class}} \]  
\[ (9-1) \]
Other measures of in-class discrimination may be more appropriate; analysis of the clustering in feature space would be one interesting area of research. Information theory also provides other ways of measuring the information conveyed by the collection of pixels from different images.

This work has been based on one possible extension of the Gabor filter to hypercomplex numbers, using the geometrically interpreted convolution. Three other possible formulations were proposed; of the three, the color opponent model may offer a richer set of results, as well as possibilities for designing the dual-color Gabor filters to suit the particular problem of face characterization.

Note that the complex Gabor as defined in previous work ([Daugm88], [Duc99], [Liu01], etc.) was a mapping from $\mathbb{R} \rightarrow \mathbb{C}$, with a consequent doubling in dimensionality. Thus, the result contains both magnitude and phase information. The Gabor filter used in this testing was a mapping from $\mathbb{Q} \rightarrow \mathbb{Q}$. The quaternionic result has a magnitude and three angles, interpreting the imaginary portion of the result as a vector in $\mathbb{R}^3$. However, an alternative approach would be to define a function that is $\mathbb{Q} \rightarrow \mathbb{O}$; a filter that, upon application to a quaternionic color image, produces an image of octonions. A rigorous mathematical examination of this alternative mapping from color images would be of general interest and not limited to Gabor filters.

Gabor filter methods for monochrome images are motivated, in part, by their superior invariance to effects of pose, illumination and expression variations. The current work did not explore the performance of the quaternionic Gabor filters with respect to these variations. The P-I-E face database from Carnegie-Mellon [Marti01] contains images of a number of subjects under varying pose, illumination and expression conditions, and could be used to measure the related efficacy of the quaternionic Gabor filter.

One of the objections to color features for face recognition is a concern that color features would exhibit significant variation due to aging, application of cosmetics and illumination source differences. This issue was not explored in my research. It would be important to
perform some investigation and testing defined to evaluate this question – using face images collected at different times, with different illumination sources and different (yet realistic) applications of cosmetics.

Some multispectral images have four related components rather than three. Face recognition is an application area where this is true. A recent approach to increase accuracy, and specifically to reduce the effect of illumination variations, is to fuse monochromatic and infrared image information [Singh04]. They report significant increase in accuracy over the use of monochrome imagery alone. A very natural extension of their concept is to represent both the scalar-valued infrared image and corresponding color images in a quaternion; the color components would be matched to the imaginary parts and the infrared component to the real part. Since the real part would be non-zero, the interesting geometric interpretation of the convolution is no longer applicable and another form of the quaternionic Gabor filter may be more appropriate.
Appendix A - Notable Face Recognition Algorithms

The following are four of the most popular algorithms for face recognition in monochrome images.

A.1 Eigenface

The eigenface algorithm was first popularized by Pentland and Turk at the MIT media laboratory [Pentl94]. It treats the entire face image (very accurately registered) as a vector, and uses Principal Component Analysis (or PCA, also known as the Karhunen-Loève transformation [Groth99]) to determine linear combinations of these pixel values that will best discriminate among a number of images. PCA is a general method of finding orthogonal axes that contain the most information about a set of sample vectors in a many-dimensional space. For face recognition, practical systems use a linear combination of the components in only the most significant axes, so the transformation determines a set of features computed from the pixel values that will adequately distinguish between the faces.

A.1.1 Basic Method

The procedure is as follows:

1) For the given training set of images \( T_1, T_2, \ldots, T_n \), subtract the mean and form the covariance matrix:

\[
\mu = \frac{1}{M} \sum T_m, \quad C = \frac{1}{M} \sum_{n=1}^{M} (T_n - \mu)(T_n - \mu)^T
\]  \hspace{1cm} (A-1)

2) Find the vector of eigenvalues \( \Lambda \) and eigenvectors \{\psi\} of \( C \).

3) Take the eigenvectors corresponding to the \( n \) largest (in magnitude) eigenvalues; these are the vectors of weights applied to the pixel values to compute the new features. Usually, \( n \) is determined by experimentation and by examination of the values of \( \Lambda \).

4) These new features are computed for unknown images, and standard distance measures are used to classify the image as one of the faces in the test set or, if the distance is too large, as unknown.
5) The eigenvectors thus found form a vector basis for the new space, and the most significant of them visually represent orthogonal sets of “characteristics” of the training set. These can be viewed as image data, and are usually called the “eigenfaces” of the training set.

![Figure 23 – the first eight of a typical set of eigenfaces](image)

The vectors and matrices in this method are large; for a 320 by 240 monochrome image, the vectors are 76800 by 1 and the matrix $C$ for which we find eigenvalues is 76800 by 76800, which is difficult computationally. However, given the over-determined nature of the matrix when the number of images in the training set $m$ is less than the number of pixels, a simplification is possible requiring only solution of an $m$-by-$m$ matrix [Turk91].

### A.1.2 Adaptations and Variations

This classic technique dictates that the transformation be recomputed for each set of face images – which would require the matrix solution any time a new enrollment is performed. In actuality, for a given image format and population, the transformation is not recomputed for each new enrollment.

One variation ([Marti01]) uses Fisher’s Linear Discriminant to determine the vectors in the space that best distinguish between the different individuals in the training set. This method has some merit, but does not perform as well when the number of enrollment images for each individual is small.

The generic nature of Principal Component Analysis has great appeal, and some have attempted to use various pre-processes to increase its robustness. For example, one promising approach [Liu03] is to apply a Gabor filter to the image at a dense grid of points, and form a vector of the resulting responses. This results in lower error rates than the pixel-based PCA, but only slightly.

### A.1.3 Advantages / Disadvantages
The eigenface method and its variants are relatively easy to understand, and have a huge amount of published research behind them. However, this approach requires very accurate registration of the face images, and is not optimized in any way for the specific characteristics of the face or typical lighting and pose issues. It is regarded in the industry as the “venerable” eigenface method.

A.1.4 Commercial Use

Some systems from Viisage use a technique called Independent Component Analysis, which is related to the eigenface process.

A.2 Local Feature Analysis

The local feature analysis method was originally based on the concepts of the eigenface. The generation of the most relevant projections of the pixel space to distinguish a population, without need of specific knowledge of the application, is powerfully simple. Yet, it was seen as a disadvantage that the eigenface method generally derived eigenvectors that were composed of scattered, seemingly unrelated pixels. Not surprisingly, these vectors were very prone to disturbance due to expression and pose changes. This led to the definition of local feature analysis, in which groups of highly relevant pixels are identified that are proximate.

A.2.1 Basic Method

Local Feature Analysis ([Penev96]) is partly motivated by evidence that humans recognize faces in parts, and also by the non-specific nature of Eigenface and other global methods. This method extracts features from the face image using a set of mappings derived by a PCA-like process, yet preserving “topography” (the spatial relationships of the pixel locations). In other words, the method attempts to find the sets of image pixels that best distinguish between the faces given, as in the eigenface approach, yet requires that these sets of pixels be close together. This results in mappings that are more intuitive visually, as can be seen in Figure 24 below.

Although these different combinations of pixels no longer form orthogonal bases for the space, they can be used to distinguish between the face images using a standard classification method.
The selection of an appropriate set of features to use is more complicated, and involves a process known as “sparsification”. The literature describes the use of a neural network to perform this sparsification.

![Figure 24 – Selection of Local Features](image)

A.2.2 Adaptations and Variations

While specific adaptations to this method are proprietary, some speculation can be done. The sparsification process allows for extensive ad-hoc tuning methods. It is likely that sparsification makes use of a priori information. As with the eigenface method, preprocessing such as spatial filtering is almost certainly done to reduce sensitivity to mislocation.

A.2.3 Advantages / Disadvantages

Anecdotal evidence suggests that these algorithms may be better than the related eigenface method at rejecting non-face objects; they are reported to be very efficient methods at match time. As with the eigenface method, the need for accurate face registration is strict: it has been stated that incorrect eye location in the image to be matched is the greatest single source of error! However, because contributors to the different features are spatially localized, the overall set is probably more tolerant of misregistration.

A.2.4 Commercial Use

FacelIt from Identix/Visionics uses Local Feature Analysis, as do the products from BioID.

A.3 Artificial Neural Networks
Neural networks have been used for recognition of many kinds of objects where *teach by example* is a possibility. Face recognition is one of these areas. Interestingly, to date neural networks have not gained the commercial success that other methods have.

**A.3.1 Basic Method**

A neural network is a set of processing elements, analogous to neurons in the brain, connected in a multi-layer fashion to perform processing of input data vectors. They were originally described in physiological research on the brain, and were later implemented to solve data classification problems.

![Neural network with two hidden layers](image)

**Figure 25 - Neural network with two hidden layers**

Typically, a neural network consists of neurons connected in several layers: one input layer, one output layer and one or more “hidden” layers. At each layer, the neuron forms a weighted sum of all of the outputs of the previous layer and transforms the sum through a non-linear function called a “squashing” function (often a “sigmoidal” function such as tanh()) since it tends to compress the output data to avoid truncation. In a classification application, the outputs of the final layer are compared and the class identified with the largest output is assigned.

The knowledge in the neural net is contained in the weights used to create the sums at the various neurons. These weights are calculated by a training process using labeled data. The most common method is “back-propagation” in which the weights are iteratively adjusted.
Appendix A – Notable Face Recognition Algorithms

based on their contribution to the output vector (computed using its partial derivatives) until all input vectors produce the desired outputs.

General problems with neural networks are:

- It is difficult to determine the proper architecture of the NN – how many layers should there be, and how many neurons in each?
- Training is very non-linear, and many ad hoc procedures are suggested to speed the convergence.
- The network can become over-trained, highly optimized to the training set but unsuited for new data vectors.
- “Regularization” can be elusive – this is the property that the NN behaves predictably for new data that is close to existing points.

A.3.2 Adaptations and Variations

Basic neural networks are not optimal for the specifics of the face recognition problem, so most real-world applications use some adaptations. AcSys uses a “Quantum Holographic Neural Network” from AND Corporation. Quantum Neural Networks ([Purus97]) substitute a more complex non-linear function for the sigmoidal function at the output of each node, usually a family of functions with different activation levels modeled on quantum energy states. The “holographic” term refers to the fact that information about any particular face is distributed throughout the network, as in a hologram. These enhancements probably give somewhat better generalization and teaching performance than a simpler neural network.

A Support Vector Machine (SVM) is an alternate method of defining classification boundaries in high-dimensional space. As any other classifier, it can be applied to either pixel values (global methods) or to more specific features: local feature appearance, inter-landmark distances or wavelet response features. As with any other classification method, the performance is dominated by the features chosen rather than the classification method. Use of a SVM for face recognition is described in [Heise01].

A.3.3 Advantages / Disadvantages
Once the training is done, a neural network classifier can be very fast (excluding any time required to compute features). A very large body of work has been done on the general area of neural networks and on the specifics of its application to face recognition.

Unfortunately, presentation of any new enrollment image has the potential to require new training of the neural net – standard methods do not provide for incremental addition of a new node in the output layer! Commercial appliers of NN technology have most certainly addressed this problem, but I am not aware of their solution.

### A.3.4 Commercial Use

AcSys and Miros have stated that their products are based on neural networks.

### A.4 Gabor Methods

Many of the most recent publications in face recognition use some method based on face feature extraction using Gabor filters. In some cases, very good performance is being reported.

#### A.4.1 Basic Method

Use of Gabor wavelets for face recognition is also biologically motivated; there is some physiological evidence that mammalian visual systems operate in part by detecting characteristics at a variety of spatial frequencies – see [Rakov01] and [Jones87]. In addition, tests have shown that features extracted by computing the filter response of a set of Gabor kernels on face images are less sensitive to lighting and pose variations than other features. However, variations in expression must still be dealt with.

The basic method is described in [Wisko99] and [Liu01]. First we define a lattice or grid of non-uniformly located points (perhaps 40 points total) on the face area, and at each point, compute the response of a family of Gabor kernels given by:

\[
G_{k,\theta}(x, y) = \left\{ \cos(\theta) - e^{-\sigma^2/2} \right\} e^{-j\frac{y}{\sigma}} + j\left\{ \sin(\theta) e^{-j\frac{x}{\sigma}} \right\}
\]

(A-2)
This complex quantity is computed for (typically) eight equally spaced rotations (θ) and five scales (k). These forty complex quantities are grouped into a feature called a jet; jets are computed at each point on the grid to form the feature representation of the face.

The Gabor jets are still sensitive to spatial mislocation, so that the jets must be accurately registered. However, even the normal spatial variations in the face are problematic, since usually the points to be encoded on the face image are located at areas of interest such as the point of the nose, corners of the mouth and so on; this process is prone to some mislocation error. The matching technique has been modified to be tolerant of some spatial variations in each point; the resulting method is called “Elastic Bunch Graph Matching”. Key to this method is to store the set of jets for a large number of face images (70 is a common value) and to use all of them in the matching.

A.4.2 Adaptations and Variations
As has been mentioned earlier, extraction of features via the Gabor filters can be used as a preprocessing step to several other techniques: PCA and neural nets, especially.

A.4.3 Advantages / Disadvantages
Test results do confirm that the Gabor features are more tolerant of illumination and pose variation. The major disadvantage of this method seems to be the processing required. Extraction of the features requires application of 40 complex filters in the neighborhoods of perhaps 40 points across the face, and matching involves many candidate templates for each person in the database. One recent implementation used a hardware accelerator board to speed up the process. These methods are more recent in development and have less (but still a significant amount) of published research behind them.

A.4.4 Commercial Use
Two companies are currently identified as using these Gabor-based techniques: ZN Bochum GmbH and Eyematic Interfaces Inc. Both of these companies have connections with the research groups at Rühr-Universitat Bochum and the University of Southern California. Cognitec Systems seems to use a form of graph-based matching, possibly using Gabor features.
Appendix B - Quaternions

Quaternions are hypercomplex numbers, discovered by Sir William Hamilton ([Hamil50]). Hamilton had been searching for a way to meaningfully represent triples of numbers, with useful and sensible definitions of the usual algebraic operations and properties. He discovered on October 16, 1843 that it was necessary to introduce a fourth dimension to preserve what he felt were the most important algebraic properties [Koets95]: modality \((a \times b = 0 \text{ only if either } a \text{ or } b \text{ are 0})\) and distributivity of multiplication over addition. He defined three purely complex quantities which are “unconnected by any linear relation with each other” [Hamil50] or with a purely real quantity.

The three complex quantities that Hamilton defined are usually written as \(i, j, k\). A quaternion can be written as the sum of its four components (one real and three imaginary) as \(q_0 + q_1i + q_2j + q_3k\), \(q \in R\). The underbar indicates that \(\bar{q}\) is a quaternion. \(i, j, k\) are related by the following key equations:

\[
\begin{align*}
i^2 &= j^2 = k^2 = -1 \\
ij &= k \\
jk &= i \\
kj &= j \\
i^2 &= -k \\
j^2 &= -i \\
k^2 &= -j
\end{align*}
\]

(B-1)

Quaternions received some attention immediately after their exposition by Hamilton, and then became somewhat obscure for over a century. When problems in three-space rotation arose in connection with orbital mechanics and astronautics, quaternion operations to express three-dimensional rotations began to receive renewed interest. More recent activity in computer graphics and image processing has further renewed interest in quaternions and their algebra.

The notion of representing a three-valued vector by a pure quaternion (a quaternion whose real part is zero) led naturally to application to color images. The image processes that have been extended to quaternions include: correlation-based pattern recognition in color images [Pei01], quaternionic Fourier transforms of color images [Sangw01], image registration [Secre01] and
quaternionic Gabor wavelets to derive local structure information on monochrome images, using the magnitude and phase information separately.

### B.1 Properties of Quaternions

Quaternions obey the following properties and theorems [Perki68]. While other formulations are possible, throughout my work and this document I assume that the quaternion components (the $q_m$ and $p_m$) are real numbers.

A general quaternion $q$ can be expressed in terms of its components as $q_0 + q_1 i + q_2 j + q_3 k$.

Quaternions add by component-wise addition:

$$ p + q = p_0 + q_0 + (p_1 + q_1)i + (p_2 + q_2)j + (p_3 + q_3)k $$

(B-2)

The product of two quaternions $p$ and $q$ can be written as:

$$ pq = (p_0 q_0 - p_1 q_1 - p_2 q_2 - p_3 q_3) + (p_0 q_1 + p_1 q_0 + p_2 q_3 - p_3 q_2)i + $$

$$ (p_0 q_2 + p_2 q_0 + p_3 q_1 - p_1 q_3)j + (p_0 q_3 + p_3 q_0 + p_1 q_2 - p_2 q_1)k $$

(B-3)

so of course (note the non-commutativity of multiplication)

$$ qp = (q_0 p_0 - q_1 p_1 - q_2 p_2 - q_3 p_3) + (q_0 p_1 + q_1 p_0 + q_2 p_3 - q_3 p_2)i + $$

$$ (q_0 p_2 + q_2 p_0 + q_3 p_1 - q_1 p_3)j + (q_0 p_3 + q_3 p_0 + q_1 p_2 - q_2 p_1)k $$

(B-4)

As can be seen from equation (B-4), reversing the order of multiplication of the imaginary components inverts the sign of the product. For this reason, quaternion multiplication is not commutative; multiplication of pure quaternions (those with no real part) is anti-commutative, but the inclusion of a real part makes the product non-commutative in any sense.

The multiplication of two pure quaternions is of special geometric interest: the real part of the result is the negative of the dot product of the vector parts of the two quaternions, while the vector part is the cross product of the two vector parts:
\[ pq = -p \cdot q + p \times q, \text{ for } p \text{ and } q \text{ pure} \]  
(B-5)

This is a specific case of the more general expression of quaternion product in dot- and cross-product form:

\[ pq = p_0q_0 - p \cdot q + p_0q + q_0p + p \times q \]  
(B-6)

### B.1.1 Conjugation

The conjugate of the quaternion \( q \), denoted by \( q^* \), is defined as \( q^* = q_0 - q_1i - q_2j - q_3k \).

Quaternion conjugation has the following properties:

\( (q^*)^* = q \) (conjugation is its own inverse operation)

\( qq^* = q^*q \) (any quaternion and its conjugate are commutative)

\( (q + p)^* = q^* + p^* \) (conjugation is distributive across addition and associative)

\( (pq)^* = q^* p^* \) (conjugation distributes by reversing the order of products)

### B.1.2 Norm

The norm (specifically the \( L^2 \) norm) of the quaternion \( q \), denoted by \( \|q\| \), is defined as \( \sqrt{qq^*} \)

\[ \|pq\| = \|p\|\|q\| \] (the norm of a product is the product of the individual norms)

\[ \|q\| = \|q^*\| \] (the norm of a quaternion is equal to the norm of its conjugate)

\[ a a^{-1} = 1, \text{ where } a^{-1} = \frac{a^*}{\|a\|} \] (the multiplicative inverse is the conjugate divided by the norm)
B.1.3 Powers

For a quaternion \( q \), whenever \( q_1^2 + q_2^2 + q_3^2 = r^2 > 0 \) and \( z = q_0 + i s \) and \( z^n = a + ib \), then
\[
q^n = a + \lambda (q_1 i + q_2 j + q_3 k), \quad \text{where } \lambda = \frac{s}{p} \text{ and } z \in Z.
\]

B.1.4 Polar Representation

Quaternions may also be represented in polar form as \( q = \|q\|e^{\mu \Phi} \), where \( \mu \) is a unit pure quaternion (real part = 0) called the eigenaxis, and \( \Phi \) is the eigenangle, \( 0 \leq \Phi \leq \pi \). The eigenaxis \( \mu \) is defined as \( \mu = \frac{\text{Im}(q)}{\|\text{Im}(q)\|} \), while the eigenangle \( \Phi \) is defined as
\[
\arctan \left( \frac{\|\text{Im}(q)\|}{\|\text{Re}(q)\|} \right) ; \text{if either denominator vanishes, the corresponding quantity is undefined.}
\]

B.2 Euler Identity for Quaternions

The generalization of the Euler identity for complex numbers to pure quaternions is
\[
e^{\theta e} = \cos(\theta) + \mu \sin(\theta).\]

Therefore, exponentiation of any general quaternion \( q \) can be determined as:
\[
e^q = e^{q_0} e^{\frac{\text{Im}(q)}{\|\text{Im}(q)\|}} = e^{q_0} e^{\left[ \frac{\text{Im}(q)}{\|\text{Im}(q)\|} \right]} = e^{q_0} \left( \cos\left(\frac{\text{Im}(q)}{\|\text{Im}(q)\|}\right) + \left( \frac{\text{Im}(q)}{\|\text{Im}(q)\|}\right) \sin\left(\frac{\text{Im}(q)}{\|\text{Im}(q)\|}\right) \right)
\]

(C-1)

If \( \mu \) is a unit pure quaternion, then \( e^{\theta e} = \left[ \cos \theta, \mu \sin \theta \right] \), where the \( [ ] \) notation indicates the real and complex parts of the quaternion: \( [\text{Re}(q), \text{Im}(q)] \). I make extensive use of this relation in developing the quaternionic Gabor filter. To see the validity of this formula, form the Taylor Series expansion of the exponential:
\[
e^{\theta e} = e^{q_0} + \sum_{k=1}^{\infty} \frac{\theta^k}{k!} \left( \frac{1}{k!} \right) = 1 + \theta q_0 + \frac{\theta^2}{2!} q_2 + \frac{\theta^3}{3!} q_3 + \frac{\theta^4}{4!} q_4 + \ldots
\]
\[
= 1 + \theta (q_0 + q_1 i + q_2 j + q_3 k)
\]
\[ + \frac{\theta^2}{2} \left( q_0^2 + q_0 q_i + q_0 q_j + q_0 q_k + q_1 i q_0 + q_1 i q_2 + q_1 i q_2 j + q_1 i q_2 k + q_2 j q_0 + q_2 j q_2 j + q_2 j q_2 k + q_2 j q_2 j + q_2 j q_2 k + q_3 k q_0 + q_3 k q_2 j + q_3 k q_2 j + q_3 k q_2 k \right) \\
+ \frac{\theta^3}{3!} q^3 + \frac{\theta^4}{4!} q^4 + .. \\
= 1 + \theta (q_0 + q_1 i + q_2 j + q_3 k) \\
+ \frac{\theta^2}{2} \left( q_0^2 - q_1^2 - q_2^2 - q_3^2 + 2q_0 q_i + 2q_0 q_j + 2q_0 q_k \right) + \frac{\theta^3}{3!} q^3 + \frac{\theta^4}{4!} q^4 + .. \\
= 1 + \theta (q_0 + q_1 i + q_2 j + q_3 k) \\
+ \frac{\theta^2}{2} \left( q_0^2 - q_1^2 - q_2^2 - q_3^2 + 2q_0 q_i + 2q_0 q_j + 2q_0 q_k \right) + \frac{\theta^4}{4!} q^4 + .. \\
+ \frac{\theta^3}{3!} \left( q_0^3 - q_0 q_i^2 - q_0 q_j^2 - q_0 q_k^2 + 2q_0^2 q_i + 2q_0^2 q_j + 2q_0^2 q_k + q_i q_0 q_i - q_i q_2 q_i - q_i q_2 q_i + 2q_i q_0 q_i + 2q_i q_0 q_j + 2q_i q_0 q_k \right) \\
+ \frac{\theta^4}{4!} (q_0^4 - q_0 q_i^2 - q_0 q_j^2 - q_0 q_k^2 + 2q_0^2 q_i + 2q_0^2 q_j + 2q_0^2 q_k + q_i q_0 q_i - q_i q_2 q_i - q_i q_2 q_i + 2q_i q_0 q_i + 2q_i q_0 q_j + 2q_i q_0 q_k) \\
= 1 + \theta (q_0 + q_1 i + q_2 j + q_3 k) \\
+ \frac{\theta^2}{2} \left( q_0^2 - q_1^2 - q_2^2 - q_3^2 + 2q_0 q_i + 2q_0 q_j + 2q_0 q_k \right) + \frac{\theta^4}{4!} q^4 + .. \\
+ \frac{\theta^3}{3!} \left( q_0^3 - q_0 q_i^2 - q_0 q_j^2 - q_0 q_k^2 + 2q_0^2 q_i + 2q_0^2 q_j + 2q_0^2 q_k + q_i q_0 q_i - q_i q_2 q_i - q_i q_2 q_i + 2q_i q_0 q_i + 2q_i q_0 q_j + 2q_i q_0 q_k \right) \\
+ \frac{\theta^4}{3!} (q_0^3 - q_0 q_i^2 - q_0 q_j^2 - q_0 q_k^2 + 2q_0^2 q_i + 2q_0^2 q_j + 2q_0^2 q_k + q_i q_0 q_i - q_i q_2 q_i - q_i q_2 q_i + 2q_i q_0 q_i + 2q_i q_0 q_j + 2q_i q_0 q_k) \\
= 1 + \theta (q_0 + q_1 i + q_2 j + q_3 k) \\
+ \frac{\theta^2}{2} \left( q_0^2 - q_1^2 - q_2^2 - q_3^2 + 2q_0 q_i + 2q_0 q_j + 2q_0 q_k \right) \\
+ \frac{\theta^3}{3!} \left( q_0^3 - q_0 q_i^2 - q_0 q_j^2 - q_0 q_k^2 + 2q_0^2 q_i + 2q_0^2 q_j + 2q_0^2 q_k + q_i q_0 q_i - q_i q_2 q_i - q_i q_2 q_i + 2q_i q_0 q_i + 2q_i q_0 q_j + 2q_i q_0 q_k \right) \\
+ \frac{\theta^4}{4!} (q_0^4 - q_0 q_i^2 - q_0 q_j^2 - q_0 q_k^2 + 2q_0^2 q_i + 2q_0^2 q_j + 2q_0^2 q_k + q_i q_0 q_i - q_i q_2 q_i - q_i q_2 q_i + 2q_i q_0 q_i + 2q_i q_0 q_j + 2q_i q_0 q_k)
\[
\begin{align*}
&+ \frac{\theta^4}{3!} (q_0^3 - q_0 q_1^2 - q_0 q_2^2 - q_0 q_3^2 - 2q_1^2 q_0 - 2q_2^2 q_0 - 2q_3^2 q_0) + \frac{\theta^3}{3!} (3q_0^2 q_1 - q_1^3 - q_1 q_2^2 - q_1 q_3^2) + \frac{\theta^2}{2} (q_0 q_1 - q_1 q_0 - q_0 q_2 + q_2 q_0 + q_0 q_3 - q_3 q_0) + \frac{\theta}{3} (q_1 - q_2 + q_3) + \frac{\theta^4}{4!} q^4 + \ldots
\end{align*}
\]

Assuming a pure quaternion, so \( q_0 = 0 \):

\[
\begin{align*}
&= 1 + \theta(q_i i + q_j j + q_k k) - \frac{\theta^2}{2} (q_0 q_1^2 + q_2^2 + q_3^2)
\end{align*}
\]

Now include the 4\(^{th}\) term:

\[
\begin{align*}
&= 1 + \theta(q_i i + q_j j + q_k k) - \frac{\theta^2}{2} (q_0 q_1^2 + q_2^2 + q_3^2)
\end{align*}
\]
Grouping by $i$, $j$ and $k$:

$$= \left(1 - \frac{\theta^2}{2}(q_1^2 + q_2^2 + q_3^2) + \frac{\theta^4}{4!}(q_1^2(q_1^2 + q_2^2 + q_3^2) + q_2^2(q_1^2 + q_2^2 + q_3^2) + q_3^2(q_1^2 + q_2^2 + q_3^2)) \right)$$

$$+ i\left(\alpha q_1 - \frac{\theta^3}{3!}q_1^3 + q_1q_2^2 + q_1q_3^2\right) + j\left(\alpha q_2 - \frac{\theta^3}{3!}q_2^3 + q_2q_1^2 + q_2q_3^2\right) + k\left(\alpha q_3 - \frac{\theta^3}{3!}q_3^3 + q_3q_1^2 + q_3q_2^2\right) + \frac{\theta^5}{5!}q^5 + ...$$

We now restrict ourselves to the case of a unit pure quaternion, $\sqrt{q_1^2 + q_2^2 + q_3^2} = 1$, so:

$$e^{2\theta} = \left(1 - \frac{\theta^2}{2} + \frac{\theta^4}{4!}\right) + i\left(\alpha q_1 - \frac{\theta^3}{3!}q_1^3\right) + j\left(\alpha q_2 - \frac{\theta^3}{3!}q_2^3\right) + k\left(\alpha q_3 - \frac{\theta^3}{3!}q_3^3\right) + \frac{\theta^5}{5!}q^5 + ..$$

Rearranging:

$$e^{2\theta} = \left(1 - \frac{\theta^2}{2} + \frac{\theta^4}{4!}\right) + \left(q_i + q_j + q_k\right)\left(\theta - \frac{\theta^3}{3!}\right) + \frac{\theta^5}{5!}q^5 + ..$$

expanding the fifth and sixth power provides additional terms:

$$e^{2\theta} = \left(1 - \frac{\theta^2}{2} + \frac{\theta^4}{4!} - \frac{\theta^6}{6!}\right) + \left(q_i + q_j + q_k\right)\left(\theta - \frac{\theta^3}{3!} + \frac{\theta^5}{5!}\right) + \frac{\theta^7}{7!}q^7 + ..$$

This is easily recognized as $e^{2\theta} = \cos \theta + q \sin \theta$, which we have shown to be valid for unit pure quaternions $q$. 
Appendix C - UML Design of Implementation

<table>
<thead>
<tr>
<th>Complex</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real : double</td>
</tr>
<tr>
<td>Imag : double</td>
</tr>
<tr>
<td>Re() : double</td>
</tr>
<tr>
<td>Im() : double</td>
</tr>
<tr>
<td>norm() : double</td>
</tr>
<tr>
<td>dot(in : Complex) : Complex</td>
</tr>
<tr>
<td>Complex(in real : double = 0, in Imag : double = 0)</td>
</tr>
<tr>
<td>~Complex()</td>
</tr>
<tr>
<td>Real() : Complex</td>
</tr>
<tr>
<td>Imag() : Complex</td>
</tr>
<tr>
<td>~Complex()</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Quaternion</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real : double</td>
</tr>
<tr>
<td>Imag : Quaternion</td>
</tr>
<tr>
<td>Re() : double</td>
</tr>
<tr>
<td>Im() : double</td>
</tr>
<tr>
<td>norm() : double</td>
</tr>
<tr>
<td>Quaternion(in r : double = 0, in imag : double = 0, in Imag : double = 0, in r : double = 0)</td>
</tr>
<tr>
<td>~Quaternion()</td>
</tr>
<tr>
<td>Real() : Quaternion</td>
</tr>
<tr>
<td>Imag() : Quaternion</td>
</tr>
<tr>
<td>~Quaternion()</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ImagePoint</th>
</tr>
</thead>
<tbody>
<tr>
<td>X : int</td>
</tr>
<tr>
<td>Y : int</td>
</tr>
<tr>
<td>ImagePoint(int X : int, int Y : int)</td>
</tr>
<tr>
<td>~ImagePoint()</td>
</tr>
<tr>
<td>Dist(int Pt, ImagePoint) : double</td>
</tr>
<tr>
<td>setX() : int</td>
</tr>
<tr>
<td>setY() : int</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ImageRect</th>
</tr>
</thead>
<tbody>
<tr>
<td>lower_right : ImagePoint</td>
</tr>
<tr>
<td>upper_left : ImagePoint</td>
</tr>
<tr>
<td>ImageRect(int X : int, int Y : int)</td>
</tr>
<tr>
<td>~ImageRect()</td>
</tr>
<tr>
<td>getCenter() : ImagePoint</td>
</tr>
<tr>
<td>getWidth() : int</td>
</tr>
<tr>
<td>getLowerRight() : ImagePoint</td>
</tr>
<tr>
<td>setCenter() in (int X : int, int Y : int)</td>
</tr>
<tr>
<td>setWidth() in (int X : int, int Y : int)</td>
</tr>
</tbody>
</table>

Figure 26 - UML Classes for Basic Types
Figure 27 - UML Classes for Jets
Figure 28 - UML Classes for Gabor Filters
Figure 29 - UML Classes for Face Model, Face Database and Candidate List
References


References


[Ljubl02] Computer Vision Laboratory, Faculty of Computer and Information Science, University of Ljubljana, Slovenia, online at: http://www.lrv.fri.uni-lj.si/facedb.html.


References


Vita

Creed Jones was born in West Virginia, USA, in 1959. Upon completing high school in Michigan, he earned the Bachelor of Science in Engineering (1980) and Master of Science in Electrical Engineering (1982) at Oakland University in Rochester, MI. He then began an 18-year career in the machine vision industry, holding positions of increasing responsibility at the General Motors Technical Center, Perceptics Corporation, Optimas Corporation and Sagem Morpho Inc. He has made significant contributions to the fields of machine vision, automated character recognition and biometrics, and is the holder of three patents.

In 2000 he left industry to study for the PhD at Virginia Tech. Creed is now an Associate Professor of Computer Science at Seattle Pacific University. He is currently the chair of the ANSI/INCITS M1.3 subcommittee for standardization of biometric data interchange formats.

In 1995 Creed married the former Jana Reagan, who is, along with their daughter Laura, the delight of his life.
Creed F. Jones III
creedj@spu.edu  /  http://myhome.spu.edu/creedj/

**Education**

**Ph.D. in Computer Engineering**, Virginia Polytechnic Institute and State University, Blacksburg, VA, 2005
- Concentration: image processing, computer vision and software engineering
- Dissertation: *Color Face Recognition using Quaternionic Gabor Wavelets*

**M.S. in Computer and Electrical Engineering**, Oakland University, Rochester, MI, 1982
- Concentrations: image processing and controls
- Advisor: Dr. J. Carroll Hill

**B.S. in Electrical Engineering**, Oakland University, Rochester, MI, 1980

**Experience**

**Associate Professor of Computer Science**, Seattle Pacific University, Seattle, WA, 2003-present
- Tenure-track teaching professor in the Department of Computer Science, with additional teaching responsibilities in the Electrical Engineering program
- Major teaching interests are programming, software engineering and image processing

**Senior Product Engineer**, Sagem Morpho Inc., a subsidiary of Sagem S.A., Tacoma, WA, 1999-present
- Represent Sagem Morpho on national and international biometrics standards development committees
- Biometric product / systems engineering, standards development lead and industry liaison

**Graduate Teaching Assistant**, Virginia Tech, Bradley Department of Electrical and Computer Engineering, Blacksburg, VA, Fall 2000 and Fall 2001
- GTA for Drs. Scott Midkiff in Fall 2000 and Richard Conners in Fall 2001, with time in the Computer Engineering Laboratory (CEL)

**Graduate Research Assistant**, Virginia Tech, Bradley Department of Electrical and Computer Engineering, Blacksburg, VA, Spring 2001
- GRA for BAE Systems under Drs. Peter Athanas and A. L. Abbott; developing algorithms for infrared image analysis on a custom DSP/FPGA platform

**Technical Director**, Avéreon Research (now Mindplay), Bellevue, WA, 1998-1999
- Co-founder and Technical Director
- Secured initial development contract, defined product line, led development efforts

- Created and led the vision systems development team; later led all technical activities
- Optimas provided software products for machine vision (XCaliper) and image analysis (Optimas), as well as complete image-based subsystems for OEM customers.

**Lead Systems Engineer**, Perceptics Corporation (a subsidiary of Northrop Grumman), Knoxville, TN, 1986-1996
- Systems and software engineering of image processing products – automated license plate reader, contact lens inspection system, food package inspection system

- In-house machine vision systems development and consultation for GM facilities nationwide

**Research Interests**

Face recognition and other image-based biometrics
- Quaternion methods

Image processing and feature extraction – color and monochrome
- Software engineering
Reviewed Publications

- Creed Jones and John Merva, *Vision-Based Process Control of Circuit Film Emboss*, Vision West, Society of Manufacturing Engineers, 1986 (received best paper award for conference)

Other Publications

*Biometrics: Identifying the Issue* (joint article), MOVE (the magazine of the American Association of Motor Vehicle Administrators), Winter 2000

Invited Presentations and Workshops

*Paying at your Fingertip: Dawning of the Biometric Age*, nacs.tech 2002 (National Association of Convenience Stores), April 23, 2002, Dallas, TX (speaker)
*Developing a BioAPI compliant application*, presented by the BioAPI consortium at the 2000 Biometric Consortium Conference, April 2000, Washington, DC (speaker and panel member)

Affiliations and Memberships

Institute of Electrical and Electronic Engineers: Computer and Signal Processing Societies (2000 - current)
ANSI/INCITS M1 Committee for biometrics standards
ANSI/INCITS M1.3 Task Group on Biometric Data Interchange Formats - chair (2001 - current)
ANSI/INCITS B10.9 subcommittee on biometrics – chair (2001)
ANSI/INCITS B10.8 biometric task force – technical editor (1999 - 2001)

Patents

US Patent #6,078,698 – “System for Reading Data Glyphs”, June 20, 2000
US Patent #5,081,685 – “Improved Method and Apparatus for Reading a License Plate”, January 14, 1992

Awards

Gilbert Faison scholarship (Virginia Tech) – 2002
William A. Blackwell Award for best research presentation (Bradley departmental award) – 2001-2002
Exceptional Technical Achievement award – Perceptics Corporation – 1991
Oakland University School of Engineering Scholarship (four years) – 1976-1981
Tau Beta Pi – Michigan Theta chapter 1979

Security clearances

US Department of Defense Secret (inactive), granted October 16, 1986
US Department of Justice Secret (inactive), granted 1986