DISSERTATION

POSE ESTIMATION OF SPHERICALLY CORRELATED IMAGES USING EIGENSPACE DECOMPOSITION IN CONJUNCTION WITH SPECTRAL THEORY

Submitted by

Randy C. Hoover

Department of Electrical and Computer Engineering

In partial fulfillment of the requirements

For the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Summer 2009

COLORADO STATE UNIVERSITY

July 8, 2009

WE HEREBY RECOMMEND THAT THE **DISSERTATION** PREPARED UNDER OUR SUPERVISION BY **RANDY C. HOOVER** ENTITLED **POSE ESTIMATION OF SPHERICALLY CORRELATED IMAGES USING EIGENSPACE DECOMPOSITION IN CONJUNCTION WITH SPECTRAL THEORY** BE ACCEPTED AS FULFILLING IN PART REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY.

Committee on Graduate Work

Professor Edwin K.P. Chong

Professor Peter Young

Professor Chris Peterson

Professor Rodney G. Roberts

Professor Anthony A. Maciejewski
Adviser

Professor Anthony A. Maciejewski
Department Head

ABSTRACT OF DISSERTATION

POSE ESTIMATION OF SPHERICALLY CORRELATED IMAGES USING EIGENSPACE DECOMPOSITION IN CONJUNCTION WITH SPECTRAL THEORY

Eigenspace decomposition represents one computationally efficient approach for dealing with object recognition and pose estimation, as well as other vision-based problems, and has been applied to sets of correlated images for this purpose. The general idea behind eigenspace decomposition is that a large set of highly correlated images can be approximately represented by a much smaller subspace. Unfortunately, determining the dimension of the subspace, as well as computing the subspace itself is computationally prohibitive. To make matters worse, this off-line expense increases drastically as the number of correlated images becomes large (which is the case when doing fully general three-dimensional pose estimation or illumination invariant pose estimation). However, previous work has shown that for data correlated in one-dimension, Fourier analysis can help reduce the computational burden of this off-line expense.

The first part of this dissertation extends some of the ideas developed for one-dimensionally correlated image data to data correlated in two- and three-dimensions making fully general three-dimensional pose estimation possible (assuming the object is illuminated from a single distant light source). The first step in this extension is to determine how to capture training images of the object by sampling the two-sphere (S^2), and the rotation group ($SO(3)$) appropriately. Therefore, a thorough analysis of spherical tessellations is performed as applied to the problem of pose estimation. An algorithm is then developed for reducing the off-line computational burden associated with computing the eigenspace by exploiting the spectral information of this spherical data set. The algorithm is based on the fact that, similar to Fourier analysis on the line or circle, spherically correlated functions can be expanded into a finite series using spherical harmonics. It is then shown that the algorithm can be extended to higher dimensions by applying a proper rotation to each of the samples defined on the surface of the sphere. Using this sampling technique, a parameterization of $SO(3)$ is

obtained. It is shown that $SO(3)$ correlated functions can be expanded into a finite series by applying a rotation to the set of spherical harmonics and expanding the function using Wigner-D matrices. Experimental results are presented to compare the proposed algorithm to the true eigenspace decomposition, as well as assess the computational savings.

The second part of this dissertation deals with the problem of pose estimation when variations in illumination conditions exist. It is shown that the dimensionality of a set of images of an object under a wide range of illumination conditions and fixed pose can be significantly reduced by expanding the image data in a series of spherical harmonics. This expansion results in a reduced dimensional set of "harmonic images". It is shown that the set of harmonic images are capable of recovering a significant amount of information from a set of images captured when both single and multiple illumination sources are present. An algorithm is then developed to estimate the eigenspace of a set of images that contain variation in both illumination and pose. The algorithm is based on projecting the set of harmonic images onto a set of Fourier harmonics by applying Chang's eigenspace decomposition algorithm. Finally, an analysis of eigenspace manifolds is presented when variations in both illumination and pose exist. A technique for illumination invariant pose estimation is developed based on eigenspace partitioning. Experimental results are presented to validate the proposed algorithm in terms of accuracy in estimating the eigenspace, computational savings, and the accuracy of determining the pose of three-dimensional objects under a range of illumination conditions.

Randy C. Hoover
Department of Electrical and Computer Engineering
Colorado State University
Fort Collins, CO 80523
Summer 2009

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

First and foremost, I wish to acknowledge my adviser, Dr. Anthony A. Maciejewski, for his constant encouragement and guidance. Dr. Maciejewski has been an integral part in my Ph.D. studies for much more that just research purposes. Dr. Maciejewski has been an excellent adviser as he has always been very open to new directions, and has steered me in the right direction when I have strayed from the path (which I do quite often). I also admire that Dr. Maciejewski has a "hands-off" approach to his graduate students allowing them the flexibility that they need, yet he was always there when I was struggling. I strongly believe that I am a better student, researcher, technical writer, and overall person because of him.

I would also like to acknowledge Dr. Rodney G. Roberts from FAMU-FSU. Dr. Roberts has been an excellent mentor and valuable resource throughout the course of this long journey. I appreciate all that I have learned from him including a better understanding of mathematics, mathematical notation, and the ability to think about problems from a new perspective. Dr. Roberts has been a constant source of inspiration throughout my Ph.D. studies.

I would like to thank my Ph.D. committee members Dr. Edwin Chong, Dr. Peter Young, and Dr. Chris Peterson for taking time out of their busy schedules to help improve my research work. I would also like thank Mr. Tom Aurand for his continued help with hardware related issues, as well as the rest of the ECE office staff for their support and encouragement.

Finally, I wish to acknowledge my wonderful wife Dr. Barbara Hoover. Her constant support through the tough times during my Ph.D. studies was invaluable. Without her constant support and encouragement, this research would never have been possible.

AUTOBIOGRAPHY

I was born in Idaho on July 16, 1975. I received my Bachelor of Science degree in Electrical Engineering and my Master of Science degree in Measurement and Control Engineering from Idaho State University in Dec. 2002 and Dec. 2004 respectively. I was a National Science Foundation Fellow from 2004 - 2005. I spent one semester at the University of Colorado at Boulder before coming to Colorado State University for my Ph.D. studies. I completed my Ph.D. in Electrical Engineering from Colorado State University in July 2009. I have joined the faculty at the South Dakota School of Mines and Technology where I am an Assistant Professor in the department of Mathematics and Computer Science. My research interests include computer vision, robotics, and control theory.

To my wife.

LIST OF SYMBOLS

Image Representation

$\mathcal{X}$: image matrix

$\mathbf{x}$: image vector

X: image data matrix

$\bar{\mathbf{x}}$: average image vector

$\bar{X}$: average image data matrix

$\hat{X}$: "unbiased" (average subtracted) image data matrix

h: number of pixels in one column of each image

v: number of pixels in one row of each image

m: total number of pixels ($h \times v$) in each image

n: total number of images in an image data set

R: sample correlation matrix

C: sample covariance matrix

Groups and Sampling

S^1: the circle group

S^2: the surface of the sphere (2-sphere)

$SO(3)$: the rotation group

α_p: angle of longitude

β_p: angle of co-latitude

γ_r: planar rotation angle

p: p^{th} sample on the surface of the sphere (S^2)

r: r^{th} planar rotation at sample p

a: number of images captured on the spheres surface

b: number of planar rotations at each of the a samples

$\boldsymbol{\xi}_p$: unit vector parameterization of S^2

$f(\boldsymbol{\xi}_p, \gamma_r)$: function sampled on $SO(3)$

Singular Value Decomposition

U: matrix of left singular vectors of X

Σ: diagonal matrix of singular values of X

V: matrix of right singular vectors of X

$\mathbf{u}_i$: the i^{th} column of U

σ_i: the i^{th} diagonal element of Σ

$\mathbf{v}_i$: the i^{th} column of V

U_k: matrix consisting of the first k columns of U

$\tilde{\mathbf{u}}_i$: estimate of $\mathbf{u}_i$

$\tilde{\sigma}_i$: estimate of σ_i

$\tilde{\mathbf{v}}_i$: estimate of $\mathbf{v}_i$

$\hat{U}$: matrix of left singular vectors of $\hat{X}$

$\hat{\Sigma}$: diagonal matrix of singular values of $\hat{X}$

$\hat{V}$: matrix of right singular vectors of $\hat{X}$

$\hat{\mathbf{u}}_i$: the i^{th} column of $\hat{U}$

$\hat{\sigma}_i$: the i^{th} diagonal element of $\hat{\Sigma}$

$\hat{\mathbf{v}}_i$: the i^{th} column of $\hat{V}$

$\hat{U}_k$: matrix consisting of the first k columns of $\hat{U}$

$\tilde{\hat{U}}$: an estimate for $\hat{U}$

$\tilde{\hat{\Sigma}}$: an estimate for $\hat{\Sigma}$

$\tilde{\hat{V}}$: an estimate for $\hat{V}$

$\tilde{\hat{U}}_k$: matrix consisting of the first k columns of $\tilde{\hat{U}}$

Quality Measures

Δ: rotation index between the two subspaces

SC: subspace criterion

ρ: energy recovery ratio

μ, ϵ: user-specified thresholds

$\|\cdot\|_F$: Frobenius norm

$\|\cdot\|_2$: 2-norm

k: dimension of the approximate eigenspace

k^*: dimension of the true eigenspace

$\mathcal{K}_i$: i^{th} Krylov subspace

Harmonic Analysis

∇_2^2: two-dimensional Laplacian

∇_3^2: three-dimensional Laplacian

∇_{2S}^2: two-dimensional spherical Laplacian

∇_{3S}^2: three-dimensional spherical Laplacian

P_l^m: associated Legendre polynomial of degree l and order m

Y_l^m: spherical harmonic of degree l and order m

$D_{m,m'}^l$: Wigner-D matrix of degree l and orders m, m'

$d_{m,m'}^l$: Wigner's (small) d-matrix of degree l and orders m, m'

Illumination Variation[1]

r: r^{th} pose of the object

$\boldsymbol{\xi}_i$: unit vector parameterization of S^2 defining the illumination direction

$\bar{X}$: matrix containing variations in pose and illumination in the spectral domain

$P_{p,q}$: associated Legendre polynomial of degree p and order q

$Y_{p,q}$: spherical harmonic of degree p and order q

$f(\boldsymbol{\xi}_i, r)$: function sampled at pose r from illumination direction $\boldsymbol{\xi}_i$

$f_{p,q}^r$: harmonic coefficient of degree p and order q at pose r

Manifold Analysis

n: total number of images

a: the total number of poses

[1]Some of the definitions in this and the next section are re-defined from above to account for variations in illumination as well as pose

b: number of illumination directions

$\mathcal{M}$: approximation to the manifold embedded in k-dimensional space

$\mathbf{c}_r$: center of mass of illumination manifold $\mathcal{I}_r$

$\mathcal{C}$: approximation to the manifold through the center of mass of $\mathcal{I}_r$ for all r

$\mathbf{p}_r^{r+1}$: orthogonal projection of $\mathcal{I}_r$ on to the linear approximation to the manifold generated from $\mathbf{c}_r$ to $\mathbf{c}_{r+1}$

$\mathbf{p}_r^{r-1}$: orthogonal projection of $\mathcal{I}_r$ on to the linear approximation to the manifold generated from $\mathbf{c}_r$ to $\mathbf{c}_{r-1}$

p_r^{r+1}: maximum element in $\mathbf{p}_r^{r+1}$

p_r^{r-1}: maximum element in $\mathbf{p}_r^{r-1}$

$\Delta\mathcal{C}$: distance between each point in $\mathcal{C}$

CHAPTER I

INTRODUCTION

1.1 Background

Object recognition and pose estimation of three-dimensional (3-D) objects from two-dimensional (2-D) images has become an important issue in computer vision applications. The recognition of 3-D objects from 2-D images has been an active research area since the early 1960s and has seen significant attention in recent years [2–4]. The general idea of object recognition is to discern information about a 3-D scene from a 2-D image. The pose estimation portion attempts to estimate the 3-D localization of the object in question. In a typical computer vision system, new incoming data (typically in the form of imaging or sensing devices) goes through a three step process. The first step involves capturing the new incoming data and converting this data to a digital format. This is typically done with a camera, range finder, thermal detection device, or any other device that allows certain information pertaining to the object to be evaluated. The second step deals with pre-processing the new data to prepare it for the final stage. The pre-processing step could involve edge detection, feature extraction, filtering, thresholding, or any number of other techniques. The third and final step of a typical computer vision system is the interpretation of the scene. This step has also been referred to as image comprehension, in which the information obtained in the previous step goes through a matching phase in an attempt to interpret information in the scene [5].

Most of the previous computer vision systems in an industrial setting have been model based [5]. Model-based object recognition methods can be categorized into three classes

based on the dimension of the spatial description of objects, i.e., 2-D, $2\frac{1}{2}$-D, and 3-D descriptions [5]. A 3-D description gives an object-centered, viewpoint-independent, and volumetric representation of the object, such as with CAD models, while a 2-D description gives a viewer-centered representation in the image space, e.g., shape features derived from the image of an object. An object description is considered $2\frac{1}{2}$-D if it is a viewer-centered representation, but depends on local surface properties of the object in each view, such as range (depth) and surface orientation [6, 7].

An alternative approach to model based recognition and pose estimation, is appearance-based methods. Appearance-based methods are not dependent on features, edges, or surface properties but instead rely only on the appearance of the object in question. Using appearance-based methods for object recognition and pose estimation is a two step process. The first step is to acquire training images of the object in question from a large number of orientations. This step attempts to capture the appearance of the object from nearly all vantage points. Once the training images have been acquired, an optimal set of images is calculated to "best" represent the training set. This first step is computed off-line and is relatively time consuming. The second step involves projecting the new incoming data onto the optimal set of images and attempting to find the best match. Once a match has been found, the object in question has been discovered and it's orientation can be calculated. The second step of the process, i.e., the on-line phase, consists of computing a matrix vector product, and then searching a manifold to find the "best" match. This phase of the object recognition and pose estimation process is very computationally efficient. Because the content of an image is affected not only by the features of objects in the image but also by environmental factors such as variation in illumination, appearance-based methods have the potential to be more robust.

This dissertation considers a class of appearance-based object recognition and pose estimation techniques referred to as subspace methods. Subspace methods were originally introduced as an efficient approach to image compression and image coding [8–10], however these techniques have also been applied to several different computer vision problems. Subspace methods have also been referred to as eigenspace methods, principal component

analysis, or the Karhunen-Loeve transformation methods [11,12]. Eigenspace methods represent a computationally efficient technique for dealing with complicated computer vision problems. As a result, eigenspace methods have been applied to several different application areas. Specific examples include face characterization and recognition [13–18], lip reading [19, 20], object recognition and pose estimation [21–31], as well as a host of applications that arise in industrial automation [32]. These applications take advantage of the fact that a set of highly correlated images can be approximately represented by a reduced dimensional set of eigenimages [24, 33].

Once the principal eigenimages of an image data set have been determined, using these eigenimages is very computationally efficient for the on-line classification of 3-D objects. Unfortunately, the off-line calculation for determining the appropriate subspace dimension, as well as the principal eigenimages themselves is computationally expensive. This drawback has been addressed using several different approaches. One approach to calculating the principal eigenimages is by using iterative techniques such as the power method [34, 35] or conjugate gradient algorithms [36, 37] that calculate one eigenimage at a time. A second approach uses the block power method and Lanczos iteration where a set of eigenimages are calculated simultaneously [38]. Another approach relies on either updating a fixed set of eigenimages by adding one new image at a time [33], or as in [39] by allowing the subspace to vary depending on the content of the new image.

A fundamentally different approach was proposed by Chang *et al.* [26] (refer to Chapter 3) where the authors show that the Fast Fourier Transform (FFT) can be used to approximate the desired subspace dimension, as well as the principal eigenimages if the image data set is correlated in one-dimension. Examples of one-dimensionally correlated image data sets given in [26] were arbitrary video sequences, as well as sequences of objects rotated through a single axis of rotation (referred to here as S^1). A modified version of Chang's algorithm has also been applied to images characterized by three parameters where the images were captured from a spherical patch above the object [40]. In [41] the computational efficiency of Chang's algorithm is increased further by exploiting the spatial coherency of the image data set as well as its spectral information.

While Chang's algorithm is the fastest known to date in terms of estimating the desired subspace dimension and approximating the resulting eigenimages, the algorithm is not directly applicable when dealing with images spherically correlated in higher dimensions. Chang's algorithm does however provide the basis for an extension to spherical correlation in higher dimensions, which is the motivation for this dissertation.

The first part of this dissertation deals with the problem of object recognition and pose estimation of 3-D objects assuming a single distant illumination source. In general, if object recognition and pose estimation of 3-D objects is desired, the training image data set needs to contain views of the object from many different orientations. A method to capture images of the object in question from a large number of orientations is presented by sampling the 2-sphere (S^2) and the rotation group ($SO(3)$) appropriately. Once the image data set has been constructed using the spherical sampling pattern, one can take advantage of the spherical correlation by using the spherical harmonic transform (SHT) to compute the spectral information (if the data is correlated on S^2). If the data is correlated on $SO(3)$, a method for computing the spectral information using spherical harmonics in conjunction with Wigner-D matrices is presented. It is also shown that pose estimation from an aerial perspective can be achieved by using a linear shifted version of the spherical harmonics referred to as hemispherical harmonics.

The second part of this dissertation deals with the problem of pose estimation assuming variations in illumination from multiple illumination sources. A method to capture images of objects under variations in both pose and illumination is presented by sampling lines of constant co-latitude (S^1), and moving the illumination sources along S^2. It is shown that the dimensionality of the data due to changes in illumination conditions and fixed pose can be significantly reduced by projecting the data onto a truncated series of spherical harmonics, generating a set of "harmonic images" at each pose. The dimensionality of the image data in the temporal dimension can then be reduced by projecting the set of harmonic images onto a set of Fourier harmonics.

In the final part of this dissertation, an analysis of eigenspace manifolds is presented when variations in illumination and pose exist. The analysis provides some insight into

the structure of eigenspace manifolds when variations in both illumination and pose are present. An alternate technique is then proposed for performing illumination invariant pose estimation.

1.2 Organization of this Study

The remainder of this dissertation is organized as follows:

Chapter 2 begins with the mathematical representation of correlated images. Eigenspace methods are then reviewed to explain how these images can be approximately represented using a small number of eigenimages. An example of eigenspace methods pertaining to pose estimation is also given. This chapter then outlines several different quality measures used in this work to compare different subspaces. A brief review of related work dealing with the eigenspace decomposition of correlated images is also given along with a summary of the contributions of the current work.

Chapter 3 gives an overview of images correlated on S^1, as well as Chang's eigenspace decomposition algorithm [6]. The chapter begins with the analysis of images obtained by a planar rotation and shows that the eigenimages can be computed in closed form using the properties of circulant matrices. This analysis is then extended to image data sets correlated on S^1, the results of which motivate the current work. Chang's algorithm as well as its computational expense is then summarized.

Chapter 4 discusses three different methods to discretize the surface of the sphere. This discretization will serve as the basis for a discrete harmonic transform detailed in Chapters 5 and 6. An analysis of the three discretizations for the purpose of pose estimation is also presented in this chapter.

Chapter 5 outlines harmonic analysis on S^2. The chapter begins by discussing harmonic analysis on S^1 and deriving the standard Fourier basis. This derivation is then extended to S^2 by introducing spherical harmonics and the associated Legendre polynomials. Based on the harmonic analysis on S^2, the spherical harmonic transform is introduced. This transform is the basis for a new algorithm for computing the eigenspace decomposition of images correlated on S^2. This eigenspace decomposition algorithm is developed in this

chapter and experimental results are presented to evaluate the accuracy of the eigenspace, the computational savings, and the energy recovered using this method. Finally, the chapter concludes with an extension to this algorithm for pose detection from an aerial perspective by replacing the standard spherical harmonic basis with a set of hemispherical harmonics.

Chapter 6 shows how the standard spherical harmonic basis can be extended to the full rotation group $SO(3)$ using Wigner-D matrices. This extension is necessary for fully general 3-D pose estimation problems. This chapter details a harmonic transform on $SO(3)$ referred to as the $SO(3)$ FFT. This development is then used to extend the algorithm developed in Chapter 5 by replacing the spherical harmonic transform with the $SO(3)$ FFT. This algorithm is then applied to fully general 3-D data sets, and an analysis is presented outlining its effectiveness in estimating the eigenspace of arbitrary 3-D objects.

Chapter 7 deals with the problem of pose estimation when variations in illumination conditions exist from both single and multiple illumination sources. It is shown that the dimensionality of a set of images of an object under a wide range of illumination conditions and fixed pose can be significantly reduced by expanding the image data in a series of spherical harmonics to obtain a set of "harmonic images". Furthermore, it is shown that the set of harmonic images are capable of recovering a significant amount of information from a set of images captured with multiple illumination sources present. An algorithm is then developed to estimate the eigenspace for a set of images that contain variation in both illumination and pose. The algorithm is based on projecting the set of harmonic images onto a set of Fourier harmonics by applying the analysis discussed in Chapter 3.

Chapter 8 proposes a technique to perform illumination invariant pose estimation. The technique is based on the fact that for most objects, variations due to a change in pose are typically higher than those due to a change in illumination conditions. Therefore, each object under a fixed pose and wide range of illumination conditions can be treated as a "class" and the problem then becomes that of determining which class new input images belong to. This chapter also provides some significant insight to the structure of eigenspace manifolds making the proposed technique possible.

Chapter 9 outlines some concluding remarks, and discusses some open research problems.

CHAPTER II

PRELIMINARIES

2.1 Chapter Overview

In this chapter, some of the fundamentals needed to apply an eigenspace decomposition to a set of correlated images is reviewed, most of which is outlined in [6,11,26]. In Section 2.2, the mathematical representation of images and image data sets is presented. The basic concept of an eigenspace decomposition is presented and the relationship between the eigenspace decomposition and the singular value decomposition (SVD) is discussed in Section 2.3. An illustrative example for pose estimation is also presented in this section. In Section 2.6, some previous work that has addressed the issue of computing the partial SVD of large matrices is presented. Finally, contributions of the current work are summarized in Section 2.7.

2.2 Mathematical Representation of Images

In this work, a gray-scale image is described by an $h \times v$ array of square pixels with intensity values normalized between 0 and 1. Thus, an image is represented by a matrix $\mathcal{X} \in [0,1]^{h \times v}$. Because sets of related images are considered here, the *image vector* $\boldsymbol{f}$ of length $m = hv$ is obtained by "row-scanning" an image into a column vector:

$$\boldsymbol{f} = \text{vec}(\mathcal{X}^T). \tag{1}$$

The *image data matrix* of a set of images $\mathcal{X}_1, \ldots, \mathcal{X}_n$ is an $m \times n$ matrix, denoted X, and defined as

$$X = [\boldsymbol{f}_1, \cdots, \boldsymbol{f}_n] \tag{2}$$

where typically $m > n$ with fixed n.

Because we will be sampling images on the sphere, it should be noted that $n = ab$, where a is the number of samples defined on the sphere's surface, and b is the number of

planar rotations captured at each sample. The image vector is then defined as

$$\boldsymbol{f} = \boldsymbol{f}(\boldsymbol{\xi}_p, \gamma_r), \tag{3}$$

where $\boldsymbol{\xi}_p$, $p \in \{0, \ldots, a-1\}$, is the unit vector pointing at the angle of co-latitude $\beta_p \in (0, \pi)$ measured down from the upper pole, and the angle of longitude $\alpha_p \in [0, 2\pi)$, which is the parameterization of the sphere in spherical coordinates. In $\boldsymbol{f}(\boldsymbol{\xi}_p, \gamma_r)$, the value $\gamma_r \in [0, 2\pi)$ is the r^{th} planar rotation at sample p where $r \in \{0, \ldots, b-1\}$.

The *average image vector* is denoted $\bar{\boldsymbol{f}}$ and defined as

$$\bar{\boldsymbol{f}} = \left[\boldsymbol{f}_1 + \boldsymbol{f}_2 + \cdots + \boldsymbol{f}_n\right]/n. \tag{4}$$

The corresponding $m \times n$ *average image data matrix*, denoted $\bar{X}$, is then defined as

$$\bar{X} = \left[\begin{array}{cccc} \bar{\boldsymbol{f}}, & \bar{\boldsymbol{f}}, & \cdots, & \bar{\boldsymbol{f}} \end{array}\right]. \tag{5}$$

Finally, the matrix $X - \bar{X}$ is computed, which is denoted $\hat{X}$ and has the interpretation of an "unbiased" image data matrix.

2.3 The Mathematical Description of an Eigenspace

Eigenspace methods have also been referred to as subspace methods, principle component analysis (PCA), or Karhunen-Loeve transformation (KLT) methods. The general idea behind eigenspace methods is to exploit correlation between images to reduce the dimensionality of the existing space. It has been shown that a set of n highly correlated images may be approximately represented by a linear combination of a set of k basis vectors, where $k < n$ [24, 26, 39]. This representation then motivates two fundamental questions:

1. Which k-dimensional subspace can best *represent* a set of n images?

2. Which k-dimensional subspace can best *distinguish* a set of n images?

The remainder of this section will show that the basis vectors in question (referred to as the eigenimages) can be defined as either the eigenvectors of the sample covariance matrix $C = \frac{1}{n}\hat{X}\hat{X}^T$ [14,24] or the eigenvectors of the sample correlation matrix $R = \frac{1}{n}XX^T$ [33,39].

2.3.1 Image Representation

In this subsection, the question of which k-dimensional subspace can best represent a set of n images is addressed. The assumptions and derivations are summarized as follows (the details can be found in [6, 11]):

Let the random vector $\boldsymbol{f}_t$ have the same statistics as the image vectors $\boldsymbol{f}_1, \boldsymbol{f}_2, \ldots, \boldsymbol{f}_n$. Then $\boldsymbol{f}_t$ can be exactly represented as a linear combination of orthonormal basis vectors $\boldsymbol{\phi}_j$ as

$$\boldsymbol{f}_t = \sum_{j=1}^{m} (\alpha_j)\boldsymbol{\phi}_j \tag{6}$$

where $\alpha_j = \boldsymbol{f}_t^T \boldsymbol{\phi}_j$. An approximation of $\boldsymbol{f}_t$, denoted $\tilde{\boldsymbol{f}}_t$, is given by

$$\tilde{\boldsymbol{f}}_t(k) = \sum_{j=1}^{k} (\alpha_j)\boldsymbol{\phi}_j + \sum_{j=k+1}^{m} b_j \boldsymbol{\phi}_j \tag{7}$$

where the b_js are constants. The "best" estimate of $\boldsymbol{f}_t$ can then be determined by defining the $\boldsymbol{\phi}_j$s and b_js to minimize the mean square error[1]

$$\begin{aligned} \bar{\varepsilon}^2(k) &= E\{\|\boldsymbol{f}_t - \tilde{\boldsymbol{f}}_t(k)\|^2\} \\ &= \textstyle\sum_{j=k+1}^{m} E\{(\alpha_j - b_j)^2\}. \end{aligned} \tag{8}$$

The optimal b_js can then be found by setting the derivative of (8) with respect to b_j to zero, which results in

$$b_j = \boldsymbol{\phi}_j^T E\{\boldsymbol{f}_t\}, \tag{9}$$

where $E\{\boldsymbol{f}_t\}$ is the expected value of the random vector $\boldsymbol{f}_t$. The mean square error in (8) then becomes

$$\begin{aligned} \bar{\varepsilon}^2(k) &= \textstyle\sum_{j=k+1}^{m} E\{(\alpha_j - b_j)^2\} \\ &= \textstyle\sum_{j=k+1}^{m} \boldsymbol{\phi}_j^T E\{(\boldsymbol{f}_t - E\{\boldsymbol{f}_t\})(\boldsymbol{f}_t - E\{\boldsymbol{f}_t\})^T\}\boldsymbol{\phi}_j \\ &\approx \textstyle\sum_{j=k+1}^{m} \boldsymbol{\phi}_j^T C \boldsymbol{\phi}_j \end{aligned} \tag{10}$$

where the covariance matrix $E\{(\boldsymbol{f}_t - E\{\boldsymbol{f}_t\})(\boldsymbol{f}_t - E\{\boldsymbol{f}_t\})^T\}$ is replaced by the sample covariance matrix $C = \frac{1}{n}\hat{X}\hat{X}^T$.

[1] Norms in all the equations in this dissertation always represent the 2-norm unless otherwise stated.

The optimal basis vectors $\boldsymbol{\phi}_j$, i.e., the eigenimages in this case, can be shown to satisfy

$$C\boldsymbol{\phi}_j = \lambda_j \boldsymbol{\phi}_j, \tag{11}$$

which are the eigenvectors of C with the corresponding eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{n-1} \geq 0$ and $\lambda_i = 0$ for $i \geq n$. (This is true because the rank of the "unbiased" image data matrix $\hat{X}$ is at most $n-1$.) The minimum mean-square error then becomes

$$\bar{\varepsilon}^2(k)_{\text{opt}} = \sum_{j=k+1}^{n} \lambda_j. \tag{12}$$

Equation (7) can be re-written in terms of the optimal $\boldsymbol{\phi}_j$s and b_js as

$$\begin{aligned} \tilde{\boldsymbol{f}}(k)_{\text{opt}} &= \textstyle\sum_{j=1}^{k} (\boldsymbol{f}^T \boldsymbol{\phi}_j)\boldsymbol{\phi}_j + \sum_{j=k+1}^{n} (\bar{\boldsymbol{f}}^T \boldsymbol{\phi}_j)\boldsymbol{\phi}_j \\ &= \bar{\boldsymbol{f}} + \textstyle\sum_{j=1}^{k} ((\boldsymbol{f} - \bar{\boldsymbol{f}})^T \boldsymbol{\phi}_j)\boldsymbol{\phi}_j. \end{aligned} \tag{13}$$

The approximation $\tilde{\boldsymbol{f}}(k)_{\text{opt}}$ is optimal for the objective function defined in (8) with $\boldsymbol{f}_t$ approximated by (7). This result shows that the entire set of image vectors $\boldsymbol{f}_1, \boldsymbol{f}_2, \ldots, \boldsymbol{f}_n$ can be "condensed" by storing the average image vector $\bar{\boldsymbol{f}}$, the first k eigenimages $\boldsymbol{\phi}_1, \boldsymbol{\phi}_2 \ldots, \boldsymbol{\phi}_k$, and the projections of $\hat{\boldsymbol{f}}_1, \hat{\boldsymbol{f}}_2, \ldots, \hat{\boldsymbol{f}}_n$ onto the first k eigenimages, where $\hat{\boldsymbol{f}}_i = \boldsymbol{f}_i - \bar{\boldsymbol{f}}$. This results in a storage of $k+1$ images versus the n original images that would need to be stored.

If one prefers to deal with the sample correlation matrix $R = \frac{1}{n} X X^T$ instead of the sample covariance matrix $C = \frac{1}{n} \hat{X} \hat{X}^T$, a related approximation of $\boldsymbol{f}_t$ is given by

$$\tilde{\boldsymbol{f}}_t(k) = \sum_{j=1}^{k} (\alpha_j)\boldsymbol{\phi}_j \tag{14}$$

which is simply the summation of the first k terms in (7), or alternatively, setting the b_j terms in (7) to zero. The corresponding mean square error then becomes

$$\begin{aligned} \bar{\varepsilon}^2(k) &= E\left\{ \|\boldsymbol{f}_t - \tilde{\boldsymbol{f}}_t(k)\|^2 \right\} \\ &= \textstyle\sum_{j=k+1}^{m} E\left\{ (\boldsymbol{f}_t^T \boldsymbol{\phi}_j)^2 \right\} \\ &= \textstyle\sum_{j=k+1}^{m} \boldsymbol{\phi}_j^T E\{\boldsymbol{f}_t \boldsymbol{f}_t^T\} \boldsymbol{\phi}_j \\ &\approx \textstyle\sum_{j=k+1}^{m} \boldsymbol{\phi}_j^T R \boldsymbol{\phi}_j \end{aligned} \tag{15}$$

where the correlation matrix $E\{\boldsymbol{f}_t \boldsymbol{f}_t^T\}$ is replaced by the sample correlation matrix $R = \frac{1}{n} X X^T$. Following the same argument as that for (11), the optimal basis vectors $\boldsymbol{\phi}_j$s, satisfy

$$R\boldsymbol{\phi}_j = \lambda_j \boldsymbol{\phi}_j \tag{16}$$

which are the eigenvectors of R.

Note that for the same subspace dimension k, equation (7) gives a better approximation than that of (14), however additional computational expense is required. In particular, equation (7) requires the additional storage of the average image vector $\bar{\boldsymbol{f}}$ as well as the subtraction of the average image vector from each image in the image data set.

2.3.2 Image Distinction

In this subsection, the question of which k-dimensional subspace can best distinguish a set of n images is addressed. Once again, the assumptions and derivations are summarized in the following (the details can be found in [6, 11]).

Consider a set of n training images, $\boldsymbol{f}_1, \boldsymbol{f}_2, \ldots, \boldsymbol{f}_n$, obtained by coarsely sampling the possible appearances of an object; and a test image $\boldsymbol{f}_t$ generated with the same statistics as the training images. In a typical pose estimation problem, the test image $\boldsymbol{f}_t$ is compared against each of the n training images in an attempt to find the best match. Let the image vector $\boldsymbol{f}_q$ be any of the n training images, then the "distance" between the two images $\boldsymbol{f}_t$ and $\boldsymbol{f}_q$ can be measured by $\|\boldsymbol{f}_t - \boldsymbol{f}_q\|^2$. Using the eigenspace representation detailed in the previous section, determining the distance between two images can be performed more efficiently by measuring $\|\tilde{\boldsymbol{f}}_t(k) - \tilde{\boldsymbol{f}}_q(k)\|^2$, where $\tilde{\boldsymbol{f}}_t(k)$ and $\tilde{\boldsymbol{f}}_q(k)$ are approximations of $\boldsymbol{f}_t$ and $\boldsymbol{f}_q$, respectively, using either (7) or (14).

For accurate pose estimation, the approximated measurement $\|\tilde{\boldsymbol{f}}_t(k) - \tilde{\boldsymbol{f}}_q(k)\|^2$ should be as close to $\|\boldsymbol{f}_t - \boldsymbol{f}_q\|^2$ as possible, in the least square sense. This requirement can be cast as an optimization problem with the objective function defined as

$$\bar{\varepsilon}^2(k) = E\{\|\boldsymbol{f}_t - \boldsymbol{f}_q\|^2 - \|\tilde{\boldsymbol{f}}_t(k) - \tilde{\boldsymbol{f}}_q(k)\|^2\} \tag{17}$$

where $\bar{\varepsilon}^2(k)$ is independent of whether or not $\boldsymbol{f}_t$ and $\boldsymbol{f}_q$ are approximated by (7) or (14).

Expanding (17) results in

$$\begin{aligned}
\bar{\varepsilon}^2(k) &= E\left\{\|\boldsymbol{f}_t - \boldsymbol{f}_q\|^2 - \|\tilde{\boldsymbol{f}}_t(k) - \tilde{\boldsymbol{f}}_q(k)\|^2\right\} \\
&= E\left\{\textstyle\sum_{j=1}^{m}(\boldsymbol{f}_t^T\boldsymbol{\phi}_j - \boldsymbol{f}_q^T\boldsymbol{\phi}_j)^2 - \sum_{j=1}^{k}(\boldsymbol{f}_t^T\boldsymbol{\phi}_j - \boldsymbol{f}_q^T\boldsymbol{\phi}_j)^2\right\} \\
&= \textstyle\sum_{j=k+1}^{m} E\left\{(\boldsymbol{f}_t^T\boldsymbol{\phi}_j - \boldsymbol{f}_q^T\boldsymbol{\phi}_j)^2\right\} \\
&= \textstyle\sum_{j=k+1}^{m} \boldsymbol{\phi}_j^T E\left\{(\boldsymbol{f}_t - \boldsymbol{f}_q)(\boldsymbol{f}_t - \boldsymbol{f}_q)^T\right\}\boldsymbol{\phi}_j \\
&= \textstyle\sum_{j=k+1}^{m} \boldsymbol{\phi}_j^T E\left\{\boldsymbol{f}_t\boldsymbol{f}_t^T + \boldsymbol{f}_q\boldsymbol{f}_q^T - \boldsymbol{f}_t\boldsymbol{f}_q^T - \boldsymbol{f}_q\boldsymbol{f}_t^T\right\}\boldsymbol{\phi}_j \\
&= \textstyle\sum_{j=k+1}^{m} \boldsymbol{\phi}_j^T\left(2E\{\boldsymbol{f}_t\boldsymbol{f}_t^T\} - 2\boldsymbol{f}\boldsymbol{f}^T\right)\boldsymbol{\phi}_j \\
&\approx \textstyle\sum_{j=k+1}^{n} 2\boldsymbol{\phi}_j^T C\boldsymbol{\phi}_j.
\end{aligned} \tag{18}$$

where the covariance matrix $E\left\{(\boldsymbol{f}_t - E\left\{\boldsymbol{f}_t\right\})(\boldsymbol{f}_t - E\left\{\boldsymbol{f}_t\right\})^T\right\}$ is replaced by the sample covariance matrix $C = \frac{1}{n}\hat{X}\hat{X}^T$. Because (18) is equivalent to (10), the optimal $\boldsymbol{\phi}_j$s are once again the eigenvectors C.

Using the above development, the distance between the test images $\boldsymbol{f}_t$ and the training images $\boldsymbol{f}_q$ can be estimated by

$$\|\boldsymbol{f}_t - \boldsymbol{f}_q\|^2 \approx \sum_{j=1}^{k}\left(\boldsymbol{f}_t^T\boldsymbol{\phi}_j - \boldsymbol{f}_q^T\boldsymbol{\phi}_j\right)^2, \tag{19}$$

and finding the best match of $\boldsymbol{f}_t$ against the training images can be approximated by

$$\min_{q=1,\ldots,n}\|\boldsymbol{f}_t - \boldsymbol{f}_q\|^2 \approx \min_{q=1,\ldots,n}\sum_{j=1}^{k}\left(\boldsymbol{f}_t^T\boldsymbol{\phi}_j - \boldsymbol{f}_q^T\boldsymbol{\phi}_j\right)^2. \tag{20}$$

Note that in computing (20), the quantity $\boldsymbol{f}_q^T\boldsymbol{\phi}_j$ is equal to $X^T\Phi_k$ where $\Phi_k = [\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \cdots, \boldsymbol{\phi}_k]$ and can be precomputed. The right-hand side of (20) can be computed in $\mathcal{O}(km) + \mathcal{O}(kn)$ flops whereas calculating the left-hand side takes $\mathcal{O}(mn)$ flops. In practical applications, $k < m$ and $k < n$, thus the minimum distance between the test image $\boldsymbol{f}_t$ against the training images is typically estimated using the right-hand side of (20).

2.3.3 Illustrative Example for Pose Estimation

This section will be concluded with a pose estimation example using the above development. Consider the image sequence shown in Fig. 1, where the top row shows five of 128 images of a boat being rotated about a single axis. The curve in the figure shows the corresponding projection of the image data matrix $\hat{X}$ onto the first three eigenimages $\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \boldsymbol{\phi}_3$, i.e., the

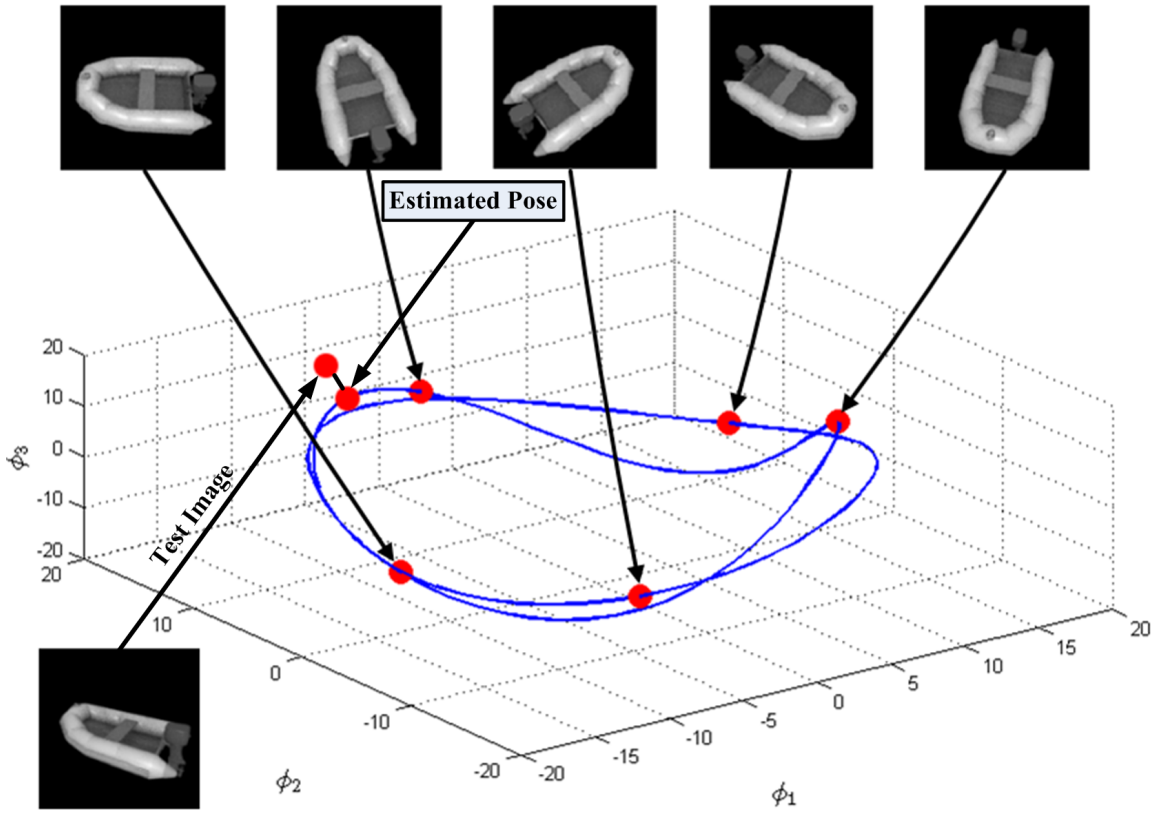

Figure 1: An example subspace in which an image data set $\hat{X}$, generated from a one-dimensional rotation, is projected onto the first three eigenimages $\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \boldsymbol{\phi}_3$. The projection generates a discrete approximation to a one-dimensional manifold embedded in three-dimensional space. The projection of five images in $\hat{X}$ onto the first three eigenimages are also shown, as well as, the projection of a test image $\boldsymbol{f}_t$ onto the three-dimensional eigenspace. The pose of the object in the test image is estimated by the closest matching point in the eigenspace.

curve shows $\Phi_k^T \hat{X}$ where $k = 3$. In general, the quantity $\Phi_k^T \hat{X}$ is a matrix of dimension $k \times n$ where each column represents a point in k-dimensional space. The collection of such points is an approximation the underlying one-dimensional manifold embedded in k-dimensional space. The problem of pose estimation then becomes that of projecting the test image $\boldsymbol{f}_t$ onto the eigenspace and determining which of the k-dimensional points is the closest. The dots in the figure show the projection of five of the original training images onto the eigenspace. A test image is also shown in the figure, along with it's projection onto the three-dimensional eigenspace. Because the test image is not contained in the original image data matrix $\hat{X}$, in general, the projection will not lie on the exactly on the approximation to

the one-dimensional manifold. Instead, the pose of the object in the test image is estimated by the pose corresponding to the closest point on the approximated manifold.

2.4 The Singular Value Decomposition

In the previous section, it was shown that the eigenimages of an image data matrix $\hat{X}$ can be computed by calculating the eigenvectors of the sample covariance matrix $C = \frac{1}{n}\hat{X}\hat{X}^T$. The eigenvectors of C provide an optimal basis in terms of image representation as well as image distinction. One immediate issue when dealing with the matrix C however, is that its construction requires multiplication of an $m \times n$ matrix with an $n \times m$ matrix. When dealing with object recognition and pose estimation problems, the value m is equal to the number of pixels in a given image, while n represents the number of images in the data set. The number of pixels is directly proportional to the resolution of the images in the data set, where the highest resolution possible is preferable in order to encapsulate as much detail as possible. This poses a problem in that even for relatively low resolution images, the number of pixels m is very large and storing the matrix C requires a large amount of memory space. For example, consider the case where each of the n images in $\hat{X}$ are of size 128×128, this results in C being of size 16384×16384. If each of the pixels in C is represented by an 8-bit number (which is typical in gray scale images), the memory requirements would be over 2Gb just to store the matrix C. Another issue is the computational cost of multiplying $\hat{X}$ with $\hat{X}^T$. This computation requires on the order of nm^2 flops, which can be computationally prohibitive.

An alternative method for computing the eigenimages of the image data matrix $\hat{X}$ is to use the singular value decomposition (SVD). The SVD of $\hat{X}$ is given by

$$\hat{X} = \hat{U}\hat{\Sigma}\hat{V}^T, \tag{21}$$

or

$$\hat{X} = \sum_{i=1}^{n} \hat{\sigma}_i \hat{\mathbf{u}}_i \hat{\mathbf{v}}_i^T \tag{22}$$

where $\hat{U} = [\hat{\mathbf{u}}_1, \hat{\mathbf{u}}_2, \ldots, \hat{\mathbf{u}}_m] \in \mathbb{R}^{m \times m}$ and $\hat{\mathbf{u}}_1, \hat{\mathbf{u}}_2, \ldots, \hat{\mathbf{u}}_m$ are referred to as the left singular vectors of $\hat{X}$. In (21), the matrix $\hat{V} = [\hat{\mathbf{v}}_1, \hat{\mathbf{v}}_2, \ldots, \hat{\mathbf{v}}_n] \in \mathbb{R}^{n \times n}$ where the vectors

$\hat{\mathbf{v}}_1, \hat{\mathbf{v}}_2, \ldots, \hat{\mathbf{v}}_n$ are referred to as the right singular vectors of $\hat{X}$. Both $\hat{U}$ and $\hat{V}$ are orthonormal, i.e., $\hat{U}\hat{U}^T = \hat{U}^T\hat{U} = I$ and $\hat{V}\hat{V}^T = \hat{V}^T\hat{V} = I$ where I is an identity matrix of appropriate size. The matrix $\hat{\Sigma} \in \mathbb{R}^{m \times n}$, with $\hat{\Sigma}^T = [\hat{\Sigma}_d \ \mathbf{0}_{n \times (m-n)}]^T$, where $\hat{\Sigma}_d = \text{diag}(\hat{\sigma}_1, \ldots, \hat{\sigma}_n)$ and $\hat{\sigma}_1 \geq \hat{\sigma}_2 \geq \cdots \geq \hat{\sigma}_n \geq 0$ are referred to as the singular values of $\hat{X}$. Note that the SVD of X is given by, $X = U\Sigma V^T$, where U and V are orthonormal, and Σ is diagonal. In (21), the $\hat{}$ notation is used to distinguish between the SVD of X and $\hat{X}$, and does not imply orthonormality.

Using the above development, the sample covariance matrix $C = \frac{1}{n}\hat{X}\hat{X}^T = \hat{U}\hat{\Sigma}\hat{V}^T\hat{V}\hat{\Sigma}\hat{U}^T = \hat{U}\hat{\Sigma}^2\hat{U}^T$, i.e., the eigenvectors of C are the left singular vectors of $\hat{X}$. The full SVD of $\hat{X}$ can be performed in mn^2 flops, which is computationally cheaper than computing the matrix C, let alone computing the eigenvectors. Alternatively, the left singular vectors of $\hat{X}$ can be obtained by using the relation $\hat{X}\hat{V} = \hat{U}\hat{\Sigma}$.

Each part of the SVD of $\hat{X}$ in (21) can be interpreted as follows: The left singular vectors (eigenimages) provide an orthonormal basis for the span of the $\boldsymbol{f}_i s$, ordered in terms of importance; the corresponding singular values measure how important the associated eigenimage is, i.e., $\|\hat{\mathbf{u}}_i^T \hat{X}\| = \hat{\sigma}_i$. The components of the i^{th} column of $\hat{V}$ measure how much each individual image contributes to the i^{th} eigenimage. The eigenspace decomposition of the sample covariance matrix C only returns the left singular vectors and singular values of $\hat{X}$. While the right singular vectors can be obtained by a simple matrix multiplication, this added expense is unnecessary when the SVD of $\hat{X}$ is computed. Although the development in the previous section only discussed the left singular vectors (eigenimages) of $\hat{X}$, the right singular vectors of $\hat{X}$ also play an important role in eigenspace decomposition algorithms. Chapter 3 will show that the right singular vectors play an important role in computing the partial eigenspace decomposition when the image sequence is correlated in one-dimension. This technique has been extended to computing the eigenspace decomposition of an image data matrix when the correlation dimensions remain orthogonal [40]. The right singular vectors are also required when computing the eigenspace decomposition of an image data matrix using it's low resolution properties [41].

2.5 Subspace Quality Measures

In practice, the singular values and the corresponding singular vectors are not known or computed exactly, and instead their estimates are used. This work addresses one computationally efficient method to estimate the first k eigenimages of an image data matrix that is correlated in multiple dimensions. Because the eigenimages are estimated, it is important to define appropriate comparison criteria that can measure the accuracy of the estimation. In general, for pose estimation problems, two questions need to be answered:

1. What subspace dimension k is needed to accurately determine the pose of a general 3-D object?

2. What measures are used to evaluate the quality of the eigenspace estimation?

In [6, 7], several error measures were discussed relating to a quantitative comparison of one subspace to another. This section reviews three of these error measures that will be used throughout this report to address both of these questions.

2.5.1 Rotation of Subspaces

The first measure discussed in this section is a measure of whether or not one subspace may be rotated into another subspace. Given two matrices of the same dimension, $A, B \in \mathbb{R}^{m\times k}$, it is desirable to know if the matrix B can be rotated into the matrix A, and if not, what is the closest it can come. This problem is formulated as follows [42]:

$$\begin{aligned} &\text{minimize:} && ||A - BQ||_F \\ &\text{subject to:} && Q^TQ = I \in \mathbb{R}^{k\times k}, \end{aligned} \tag{23}$$

where $||\cdot||_F$ represents the Frobenius norm and $\Delta = \min_Q ||A-BQ||_F$ is the residue. Because $\text{trace}(C^TC) = \|C\|_F^2$, if Q is orthogonal then

$$\begin{aligned} ||A - BQ||_F^2 &= \text{trace}(A^TA) + \text{trace}(B^TQ^TQB) - 2\text{trace}(Q^TB^TA) \\ &= \text{trace}(A^TA) + \text{trace}(B^TB) - 2\text{trace}(Q^TB^TA). \end{aligned} \tag{24}$$

Thus, (23) is equivalent to maximizing $\text{trace}(Q^TB^TA)$. The maximizing Q can be found by calculating the SVD of $C = B^TA$. Thus,

$$\text{trace}(Q^TC) = \text{trace}(Q^TU_c\Sigma_cV_c^T) = \text{trace}(Z\Sigma_c) = \sum_{i=1}^{k} z_{ii}\sigma_{ci} \leq \sum_{i=1}^{k} \sigma_{ci}, \tag{25}$$

where $Z = V_c^T Q^T U_c$, U_c and V_c are the matrices containing the left and right singular vectors of C, respectively, while Σ_c is a diagonal matrix containing the singular values of C in descending order. Thus, the upper bound is attained by setting $Q = U_c V_c^T$.

The steps for computing the Q that minimizes $||A - BQ||_F$, or alternatively, maximizes trace$(Q^T B^T A)$, can be calculated as follows:

- Form the matrix $C = B^T A$.
- Compute the SVD of C, i.e., $C = U_c \Sigma_c V_c^T$.
- Compute the orthogonal matrix $Q = U_c V_c^T$.

The residue Δ after solving (23) using the above Q will satisfy

$$\Delta^2 = \text{trace}(A^T A) + \text{trace}(B^T B) - 2\sum_{i=1}^{k} \sigma_{ci}. \tag{26}$$

The smaller the residue Δ, the closer A and B are to representing the same subspace. To determine the rotation between a set of estimated and true eigenimages, one would set

$$\begin{aligned} A &= \hat{U}_k \quad \text{and} \\ B &= \tilde{\hat{U}}_k, \end{aligned}$$

where $\hat{U}_k$ and $\tilde{\hat{U}}_k$ are the matrices containing the first k true and estimated eigenimages as their columns, respectively.

The subspaces containing true and estimated right singular vectors can then be compared using (26) and (23). If these two subspaces have orthonormal columns, then trace$(A^T A) =$ trace$(B^T B) = k$ and the residue in (26) can be given by

$$\Delta^2 = 2(k - \sum_{i=1}^{k} \sigma_{ci}) \tag{27}$$

where σ_{ci} is the i^{th} singular value of C. Note that if A and B are orthogonal to each other, then Δ achieves a worst case upper bound of $\sqrt{2k}$. Therefore, the residue is sometimes normalized by this value resulting in a upper bound of one.

2.5.2 Energy Recovery Ratio

True and estimated eigenimages of $\hat{X}$ can also be compared in terms of their capability of recovering the energy in $\hat{X}$. If the estimated eigenimages are ordered as per their importance, the first eigenimage ($\tilde{\hat{\mathbf{u}}}_1$) will give the following inequality:

$$\begin{aligned} \|\tilde{\hat{\mathbf{u}}}_1^T \hat{X}\|^2 &= \|\sum_{j=1}^n \left(\hat{\sigma}_j \tilde{\hat{\mathbf{u}}}_1^T \hat{\mathbf{u}}_j \hat{\mathbf{v}}_j^T\right)\|^2 \\ &= \sum_{j=1}^n (\hat{\sigma}_j \tilde{\hat{\mathbf{u}}}_1^T \hat{\mathbf{u}}_j)^2 \|\hat{\mathbf{v}}_j^T\|^2 \\ &= \sum_{j=1}^n (\hat{\sigma}_j \tilde{\hat{\mathbf{u}}}_1^T \hat{\mathbf{u}}_j)^2 \\ &\leq \sigma_1^2. \end{aligned} \tag{28}$$

Note that the maximum of $\|\tilde{\hat{\mathbf{u}}}_1^T \hat{X}\|^2$ is achieved when $\tilde{\hat{\mathbf{u}}}_1 = \hat{\mathbf{u}}_1$. This suggests a more general measure, i.e., the "energy recovery ratio," for estimated eigenimages of $\hat{X}$, which is defined as

$$\rho(\hat{X}, \tilde{\hat{U}}_k) = \frac{\sum_{i=1}^k \|\tilde{\hat{\mathbf{u}}}_i^T \hat{X}\|^2}{\|\hat{X}\|_F^2} \tag{29}$$

where $\tilde{\hat{U}}_k$ denotes the first k estimated eigenimages of $\hat{X}$ [6]. From the theory of principle component analysis, the true eigenimages yield the highest energy recovery ratio. Therefore, for a given energy recovery ratio ρ, the quality of the estimated eigenimages can be obtained by comparing $k - k^*$, where k^* is the subspace dimension obtained by the true eigenimages. This measure may also be used to determine the required subspace dimension k to achieve a user specified accuracy, e.g. given a user specified energy recovery ratio μ, determine k such that $\rho(\hat{X}, \tilde{\hat{U}}_k) \geq \mu$.

A related measure used in this report that is useful in determining the subspace dimension k is the change in ρ, which is defined as

$$\Delta\rho(\hat{X}, \tilde{\hat{U}}_k) = \rho(\hat{X}, \tilde{\hat{U}}_k) - \rho(\hat{X}, \tilde{\hat{U}}_{k-1}). \tag{30}$$

The change in ρ determines how much additional energy is recovered by adding the k^{th} estimated eigenimage to the subspace.

2.5.3 Subspace Criterion

The energy recovery ratio defined in (29) achieves a maximum when $\tilde{\hat{\mathbf{u}}}_i = \hat{\mathbf{u}}_i$ for $i = 1, \ldots, k$. Therefore for any given energy recovery ratio, it is possible that $k \geq k^*$, where k is the

subspace dimension as computed using the estimated eigenimages and k^* is the subspace dimension as computed using the true eigenimages. Hence another measure used in this study is the degree to which estimated eigenimages span the subspace of the first k^* true eigenimages, which will be referred to as the subspace criterion, SC, given by

$$\mathrm{SC} = \sqrt{\frac{1}{k^*} \sum_{i=1}^{k} \sum_{j=1}^{k^*} (\tilde{\hat{\mathbf{u}}}_i^T \hat{\mathbf{u}}_j)^2}. \tag{31}$$

If $\hat{U}_{k^*} = [\hat{\mathbf{u}}_1, \hat{\mathbf{u}}_2, \cdots, \hat{\mathbf{u}}_{k^*}]$ is the matrix consisting of the first k^* true eigenimages, and $\tilde{\hat{U}}_k = [\tilde{\hat{\mathbf{u}}}_1, \tilde{\hat{\mathbf{u}}}_2, \cdots, \tilde{\hat{\mathbf{u}}}_k]$ is the matrix consisting of the first k estimated eigenimages, then $\mathrm{SC} = 1$ when $\mathrm{span}(\tilde{\hat{U}}_k) = \mathrm{span}(\hat{U}_{k^*})$ and is less than 1 otherwise [7].

2.6 Related Work

2.6.1 Introduction

The principal calculation required in eigenspace methods is the precomputation of estimates of the left singular vectors $\tilde{\hat{U}}_k$ of the $m \times n$ matrix $\hat{X}$. Once the principal eigenimages of an image data matrix have been determined, using these eigenimages is very computationally efficient for the on-line classification of 3-D objects. Unfortunately, the off-line calculation for determining the appropriate subspace dimension, as well as the principal eigenimages themselves is computationally expensive. Reducing this computational expense by taking advantage of the fact that only the principle singular vectors are of interest has been a major topic of previous research. The remainder of this section outlines some of the more popular techniques used in computing the partial SVD of large matrices, most of which can be found in [6, 7].

2.6.2 Iterative Methods

One class of techniques for computing the SVD of X relies on finding the eigenimages iteratively. One such technique, referred to as the power method, calculates the dominant singular values and vectors one at a time [34, 42–44]. The algorithm is relatively easy to implement: starting with a random vector $\mathbf{v}^{(0)}$, perform the iteration

$$\mathbf{u}^{(k+1)} = \frac{X\mathbf{v}^{(k)}}{\|X\mathbf{v}^{(k)}\|} \tag{32}$$

and

$$\mathbf{v}^{(k+1)} = \frac{X^T \mathbf{u}^{(k+1)}}{\|X^T \mathbf{u}^{(k+1)}\|} \tag{33}$$

until $\|\mathbf{u}^{(k+1)} - \mathbf{u}^{(k)}\|$ drops below a preset threshold. To obtain the next pair of singular vectors, the singular vectors calculated in the previous stage are removed from X. That is, the matrix X is updated as $X' = X - \sigma_1 \hat{\mathbf{u}}_1 \hat{\mathbf{v}}_1^T$, and the same iteration is repeated with X' to find σ_2, $\hat{\mathbf{u}}_2$ and $\hat{\mathbf{v}}_2$. In [35], Shlien modified this power algorithm slightly, so that it will have better convergence properties. The cost of both these methods is on the order of $mnki$ where k is the desired dimension of the eigenspace and i is the average number of iterations needed for each pair of singular vectors [7].

There are other iterative methods that work on symmetric matrices [37] that have been applied to either $X^T X$ or XX^T. However as mentioned earlier, neither approach is practical when m and n are large.

2.6.3 Gradient Descent Algorithms

The gradient-type algorithms [36, 37] recast the search for the dominant singular vectors into an optimization problem. By definition, the left singular vector of X associated with the largest singular value is the unit vector that maximizes $\|X^T \mathbf{e}\|$, therefore the Rayleigh quotient $F(\mathbf{e})$, defined as

$$F(\mathbf{e}) = \frac{\|X^T \mathbf{e}\|^2}{\|\mathbf{e}\|^2}, \tag{34}$$

is maximized when $\mathbf{e}$ is colinear with $\mathbf{u}_1$. The search for the maximum of $F(\mathbf{e})$ is through gradient or conjugate gradient methods. The cost of each iteration is on the order of mn, therefore the total cost is on the order of $mnki$ [6, 7].

2.6.4 Block Power Methods and the Lanczos Algorithm

In [38], Vogel *et al.* considered the block power method and the Lanczos method to solve the SVD of ill-posed problems, i.e., large matrices with rapidly decaying singular values. The block power method, also known as simultaneous iteration, is similar to the power method except that it iterates with k pairs of singular vectors instead of with one pair at a time. The computational expense for this method is also on the order of $mnki$ [6].

The Lanczos method is a different approach to this problem. Let A be a symmetric matrix and $\boldsymbol{q}_0$ be an initial unit vector, then each iteration i of the Lanczos method can be viewed as a projection of the matrix A onto the i^{th} Krylov subspace

$$\mathcal{K}_i(\boldsymbol{q}_0) = \left\langle \boldsymbol{q}_0, A\boldsymbol{q}_0, \ldots, A^{i-1}\boldsymbol{q}_0 \right\rangle . \tag{35}$$

The matrix representing this projection is a symmetric $k \times k$ tridiagonal matrix T_i, where the eigenvalues of T_i converge rapidly to the extremal eigenvalues of A and the corresponding eigenvectors of T_i can be used to compute approximate eigenvectors of A. For a non-symmetric matrix X with size $m \times n$, one can apply the Lanczos method to the symmetric matrix

$$A = \begin{bmatrix} 0_{m\times m} & X \\ X^T & 0_{n\times n} \end{bmatrix} . \tag{36}$$

Note that forming A as given by (36) requires at least twice the memory space of storing X. The code listed in [38] shows a way of applying the Lanczos method to a non-square matrix X without forming the matrix A as in (36). The computational expense for this method is on the order of $mnki$ [6].

2.6.5 Eigenspace Updating

Another class of techniques relies on updating a small set of eigenimages by recursively adding one image at a time. Murakami *et al.* [33] illustrated a method for updating a fixed number of eigenimages. If the total number of images is n and the desired number of eigenimages is k, then the eigenspace decomposition of the first $k+1$ images is calculated and the first k eigenimages are kept. Then one image is added at a time to update the k eigenimages. The updating can be done efficiently by taking advantage of the orthogonality of the eigenimages from the previous stage. The cost of this method is on the order of mnk^2 and hence it has an advantage over the direct SVD algorithm when k^2 is smaller than n.

Chandrasekaran et al. [39] took a similar approach to that of [33]. The major difference is that the number of eigenimages calculated is adaptively changed. Instead of keeping only the first k eigenimages in each iteration, it is suggested to keep the eigenimages with the corresponding eigenvalues higher than a preset threshold. Also, when adding one image

that does not change the eigenimages appreciably, the next one or several images may be skipped. The authors claimed that this method is as efficient as that of [33] when the required dimension of eigenimages is small.

2.6.6 Spectral Transform Techniques

2.6.6.1 Discrete Cosine Transform

An efficient method for quickly computing an approximate value for $X^T X$ was developed by Murase *et al.* in [45]. (Recall that the eigenvectors of $X^T X$ are the right singular vectors of X and the left singular vectors of X can be easily computed from the corresponding right singular vectors.) The discrete cosine transform (DCT) was applied to blocks of each image. The approximated matrix $X^T X$ and its eigenvectors were found in the frequency domain (DCT), and then the inverse DCT was applied to the eigenvectors to transform them back to the spatial domain. This method is referred to as the spatial temporal adaptive method (STA). The number of multiplications required for applying this method to an $m \times n$ matrix to find the first k eigenimages is given by:

$$N_{STA} = 1.25mn + nm(1+\alpha^2)\beta_a + kin^2 + mn(1+k\alpha)\beta_b + 1.25kn, \tag{37}$$

where i is the average number of iterations, and α, β_a and β_b are constants. For their implementation, the authors reported that the speed is 6 to 10 times faster than the direct SVD algorithm[2] for calculating the first 8 eigenimages from a set of 256 images [6, 7].

2.6.6.2 Chang's Algorithm

The major contribution of this dissertation is motivated by the work of Chang *et al.* [26] where a fundamentally different algorithm was proposed. Because this algorithm is the motivation behind the current work, an overview will be given here, but the details can be found in Chapter 3. Chang's algorithm was motivated by the observation that for a set of planar rotated images, i.e., the i^{th} image of a set of n images is obtained from the first image by a planar rotation of $360(i-1)/n$ degrees, the matrix $X^T X$ is a "circulant matrix". The (unordered) SVD of X for this case is known in closed form, where the right singular

[2]The authors did not specify which algorithm they used to implement the direct SVD.

vectors are pure sinusoids and the left singular vectors can be calculated by applying the FFT to the rows of X. For objects correlated on S^1, as well as arbitrary video sequences, it was found that the first k estimated eigenimages can be found using only a small number of harmonics, i.e., not much larger than k. The cost of Chang's algorithm is on the order of $mn \log_2 n$, which compares favorably with most direct SVD algorithms.

In [7, 41] Saitwal *et al.* extended Change's algorithm to exploit the spatial coherency as well as the spectral aspects of the right singular vectors. The main contributions of their work were threefold, 1) the determination of the effect of resolution reduction techniques on the resulting eigenspace decomposition, 2) the appropriate method to down-sample the data in the spatial domain, and 3) the appropriate extension to Chang's algorithm to increase the computational efficiency. It was shown that random sampling outperformed any low-pass filtering techniques in reducing the resolution in the spatial domain. It was determined that using the resulting low-resolution right singular vectors to compute the eigenimages proved far better than simply "scaling" the low-resolution eigenimages themselves. Because Saitwal's algorithm also dealt with right singular vectors as opposed to the eigenimages themselves, the extension to Chang's eigenspace decomposition algorithm was immediate. Saitwal's algorithm was originally applied to arbitrary video sequences in [41], but it was shown in [7] that this technique also performs well on objects correlated on S^1. Chang's eigenspace decomposition algorithm has also been extended to data correlated in higher dimensions where the image data set was generated by capturing images of objects from a spherical patch above the object [40]. Capturing images from a spherical patch above the object allows the sampling dimensions to remain orthogonal and thus an FFT may be applied to each dimension. Unfortunately, the size of the patch must be limited to maintain orthogonality resulting in a limited number of views of the object of interest.

2.7 Summary of Current Contributions

Currently Chang's algorithm with Saitwal's extension appears to be one of the fastest known algorithms for estimating the first k eigenimages of an image data set correlated on S^1. Unfortunately, neither technique is directly applicable when dealing with objects

correlated in higher dimensions such as S^2 or $SO(3)$. This is based on the fact that both algorithms require the right singular vectors of $\hat{X}$ to be approximately spanned by sinusoids. For objects correlated on S^2 and $SO(3)$ however, this is not the case.

Moving from correlation on S^1 to correlation on S^2 and finally $SO(3)$ is inherently more difficult. This difficulty is due to the fact that for fully general 3-D pose estimation, the objects in question must be sampled from a very large number of orientations in an attempt to capture all possible vantage points. This results in very large image data sets, where the sampling dimensions are no longer orthogonal. In the first part of this work, capturing images of object from a large number of orientations using a single distant illumination source is accomplished by sampling the sphere appropriately. Three different spherical sampling methods are evaluated, namely, the Hierarchical Equal Area isoLatitude Pixelization (HEALPix) sampling pattern, sampling on an equi-angular grid of Chebychev nodes, and using a Gauss-Legendre grid. It is shown that for pose estimation problems, using the HEALPix sampling pattern is superior to the other two.

Once the image data matrix has been constructed using the above mentioned sampling method, it is shown that for images correlated on S^2, the SHT is useful in computing the harmonic power spectra. An efficient eigenspace decomposition algorithm is then presented based on the fact that most of the energy of an S^2 correlated image data set is concentrated around the low frequency spherical harmonics. It is also shown that with little modification, pose estimation from an aerial perspective can be achieved by sampling the upper hemisphere of S^2. Using this hemispherical sampling pattern the spherical harmonics can be linearly shifted to compute the frequency information of this hemispherically correlated image data set. The linear shifted versions of the spherical harmonics are referred to as the hemispherical harmonics.

Computing the spherical harmonic power spectra for data correlated on S^2 gives significant insight to the fully general 3-D pose estimation problem. It is shown that if the rotation group $SO(3)$ is sampled appropriately using the HEALPix sampling pattern, then Wigner-D matrices can be used in conjunction with spherical harmonics to compute the frequency information of this $SO(3)$ correlated image data set. Similar to data correlated on S^2, most

of the energy of the $SO(3)$ correlated data is concentrated around the low frequency $SO(3)$ harmonics. A computationally efficient algorithm for computing the eigenspace decomposition of this data set is then presented based on the spectral information as computed by the $SO(3)$ FFT.

The second part of this work deals with the problem of illumination invariant pose estimation. It is shown that images of an object viewed from a fixed pose under a wide range of illumination conditions arriving from a single illumination source but different directions can be approximated by a low-dimensional linear subspace. It is shown that this subspace can be computed by projecting the image data onto a truncated set of spherical harmonics producing a set of harmonic images. This set of harmonic images (after orthogonalization) are very close to the true eigenimages as computed using the SVD. Furthermore, it is shown that a nine-dimensional subspace is sufficient to recover over 95% of the energy for most objects. It is then shown that the subspace computed assuming a single illumination source is sufficient to recover a significant amount of energy from images captured with multiple illumination sources present. It is shown that the major effect multiple illumination sources have is an increased probability that a local specularity may be illuminated and as such, the specularity is not well represented by the low-dimensional subspace. Using the reduced dimensional subspace for objects under a wide range of illumination conditions and fixed pose, an algorithm is then developed to compute the eigedecomposition when variation in pose and illumination exist. The algorithm is based on projecting the set of harmonic images onto a set of Fourier harmonics by applying Chang's algorithm temporally.

Finally, an analysis of eigenspace manifolds is presented when variations in pose and illumination are present. This analysis reveals that for most objects, variations due to a change in pose tend to be much larger than those due to a change in illumination. Based on this analysis, a technique is presented to perform illumination invariant pose estimation.

CHAPTER III

CORRELATION ON S^1 - CHANG'S ALGORITHM

3.1 Chapter Overview

As indicated earlier, Chang's eigenspace decomposition algorithm [26] is one of the fastest known algorithms for computing the first k estimated eigenimages of correlated images to a user-specified accuracy. This chapter gives an overview of that algorithm, along with an analysis of its computational efficiency. Section 3.2 outlines how sampling lines of constant latitude results in a one-dimensionally correlated image data set that is correlated on S^1. Section 3.3 discusses a special case where the eigenspace decomposition is known in closed form. This special case provides the motivation for the extension to computing the eigenspace decomposition of image data sets correlated on S^1. This extension, as well as Chang's eigenspace decomposition algorithm, is outlined in Section 3.4.

3.2 Introduction

Consider capturing images on the surface of the sphere where the object is placed at the sphere's center, as seen in Fig. 2. Recall that the image vector is then defined as $\boldsymbol{f}(\boldsymbol{\xi}_p, \gamma_r)$, where $\boldsymbol{\xi}_p$, $p \in \{0, \ldots, a-1\}$, is the unit vector pointing at the angle of co-latitude $\beta_p \in (0, \pi)$ measured down from the upper pole, and the angle of longitude $\alpha_p \in [0, 2\pi)$, which is the parameterization of the sphere in spherical coordinates. In $\boldsymbol{f}(\boldsymbol{\xi}_p, \gamma_r)$, the value $\gamma_r \in [0, 2\pi)$ is the r^{th} planar rotation at sample p where $r \in \{0, \ldots, b-1\}$. Using this definition, and setting $\gamma_r = 0$, β_p to a constant, and $\alpha_p \in [0, 2\pi)$ constrains the sampling of the object to a line of constant latitude on the sphere. Sampling the object in this manner results in a one-dimensionally correlated image data matrix that is correlated on S^1.

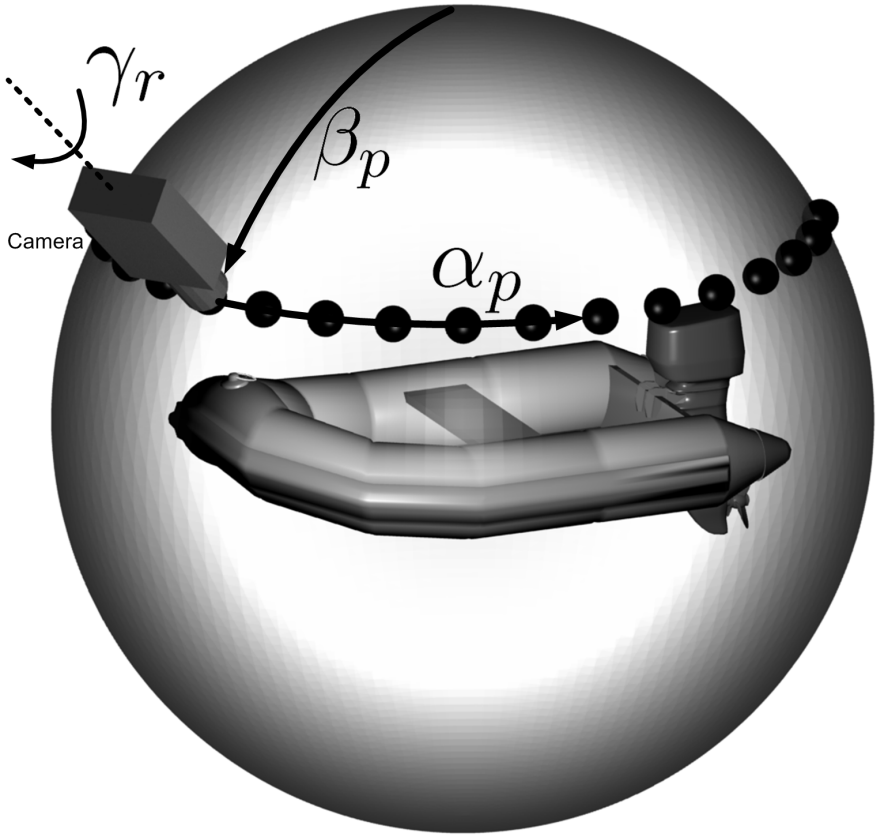

Figure 2: Sampling an object along a line of constant co-latitude, which results in an image data matrix correlated on S^1. A sample image is captured at each of the black dots that reside on the sphere.

3.3 Special Case

Before discussing the details of Chang's algorithm, a special case that motivated the development of Chang's algorithm is outlined. In the above sampling method, if α_p and β_p are set to constants, and $\gamma_r = 2\pi r/n$ with $r \in \{0, \ldots, n-1\}$, then $\boldsymbol{f}_{i+1}$ may be obtained from $\boldsymbol{f}_i$ by a planar rotation of $\gamma = 2\pi/n$. The correlation matrix $X^T X$ is then given by

$$X^T X = \begin{bmatrix} \boldsymbol{f}_1^T \boldsymbol{f}_1 & \boldsymbol{f}_1^T \boldsymbol{f}_2 & \cdots & \boldsymbol{f}_1^T \boldsymbol{f}_n \\ \boldsymbol{f}_2^T \boldsymbol{f}_1 & \boldsymbol{f}_2^T \boldsymbol{f}_2 & \cdots & \boldsymbol{f}_2^T \boldsymbol{f}_n \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{f}_n^T \boldsymbol{f}_1 & \boldsymbol{f}_n^T \boldsymbol{f}_2 & \cdots & \boldsymbol{f}_n^T \boldsymbol{f}_n \end{bmatrix}. \tag{38}$$

It can be shown [26] that X^TX is a *circulant* matrix with circularly symmetric rows. For this special case, the eigenspace decomposition of X^TX is given by the Discrete Fourier Transform (DFT), i.e.,

$$X^TX = HDH^T \tag{39}$$

where D is the $n \times n$ matrix given by

$$D = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n) \tag{40}$$

and H is the real DFT matrix defined as

$$H = \sqrt{\frac{2}{n}} \begin{bmatrix} \frac{1}{\sqrt{2}} & c_0 & -s_0 & c_0 & -s_0 & \cdots \\ \frac{1}{\sqrt{2}} & c_1 & -s_1 & c_2 & -s_2 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \cdots \\ \frac{1}{\sqrt{2}} & c_{n-1} & -s_{n-1} & c_{2(n-1)} & -s_{2(n-1)} & \cdots \end{bmatrix} \tag{41}$$

with $c_k = \cos(2\pi k/n)$ and $s_k = \sin(2\pi k/n)$.

Using the above development, an unordered SVD of X can be obtained for this special case by letting $V = H$, i.e., the right singular vectors of X in this case are given by pure sinusoids of frequencies that are multiples of $2\pi/n$ radians. The left singular vectors (eigenimages) are then computed using the relationship $XH = U\Sigma$, which can be computed efficiently using the FFT [26].

3.4 Chang's Eigenspace Decomposition Algorithm

While the above analysis does not hold for arbitrary image data sets, it has been shown in [26] that the analytical expressions for planar rotations serve as a good approximation for the eigenspace decomposition of image data sets correlated on S^1. To explore the effects of moving from planar rotations to correlation on S^1, a coffee mug was sampled as mentioned above, and its eigenspace decomposition was then computed. Fig. 3 shows five of the $n = 128$ images that make up the image data matrix X, as well as the singular vectors of X. In examining the figure, even though the results of planar rotation do not apply here, the following two properties can be observed [26, 41].

1. The right singular vectors of X are well-approximated by sinusoids of frequencies that are multiples of $2\pi/n$ radians, and the power spectra of the right singular vectors consist of a narrow band around the corresponding dominant harmonics.

2. The dominant frequencies of the power spectra of the (ordered) singular vectors increase approximately linearly with their index.

These two properties indicate that the right singular vectors of an image data set correlated in S^1 are approximately spanned by the first few low frequency harmonics. Therefore, by projecting the image data set X onto these first few low frequency harmonics, the computational expense associated with computing the SVD can be significantly reduced.

Chang's algorithm makes use of the above two properties to estimate the subspace dimension k as well as the principle eigenimages $\tilde{U}_k$ of the image data matrix X. It was shown in [26] that if the power spectra of the first q right singular vectors of X are restricted to the band $[0, 2\pi q/n]$, then for $\rho(X^T, H_q) \geq \mu$, the quantity $\rho(X, \tilde{U}_k)$ will exceed μ for some $k \leq q$, where H_q contains the first q column of H, and μ is a user specified value. This inequality shows that the energy recovery ratio as computed using the first few low frequency harmonics of H provides a lower bound on the energy recovery ratio as computed using the estimated eigenimages. Furthermore, this bound is shown to be extremely tight in most cases [26], with a tight upper bound given by the energy recovery ratio as computed by the "true" eigenimages. In other words, the first k estimated eigenimages $\tilde{U}_k$ of the matrix product XH_q are shown to be very good estimates of U_k.

The entire algorithm developed in [26] for computing the eigenspace decomposition of X is now summarized as follows:

1. Form the matrix Y whose i^{th} row is the FFT of the i^{th} row of X.

2. Determine the smallest number q such that $\rho(X^T, H_q) \geq \mu$, where μ is the user specified energy recovery ratio.

3. Let Z_q denote the matrix XH_q, compute the SVD of Z_p. The key observation here is that the matrix XH_q can be constructed from the first q columns of the matrix

$\sqrt{(2/n)}[1/\sqrt{2}\mathbf{y}_0\ \Re\mathbf{y}_1\ \Im\mathbf{y}_1\ \Re\mathbf{y}_2\ \Im\mathbf{y}_2\ \cdots]$, where $\mathbf{y}_i$ denotes the i^{th} column of Y, and $\Re$ and $\Im$ give the real and imaginary parts respectively.

4. Return $\tilde{U}_k$ such that $\rho(X, \tilde{U}_k) \geq \mu$.

The above algorithm gives the partial SVD of the image data matrix X, the eigenspace decomposition of $\hat{X}$ can be calculated with two simple modifications. First, in step 2, determine q as the smallest number such that $\sum_{i=2}^{q} \|X^T\mathbf{h}_i\|^2 \geq \mu(\|X\|_F^2 - \|X^T\mathbf{h}_1\|^2)$. Second, in step 3, the SVD of the matrix comprising the second through q columns of Z is computed [26]. If $q << n$, then Chang's eigenspace decomposition algorithm scales to approximately $\mathcal{O}(mn\log_2(n))$ which compares favorably to the $\mathcal{O}(mn^2)$ flops required by most SVD algorithms.

While Chang's algorithm provides an efficient method to compute the eigenspace decomposition of an image data set correlated on S^1, the algorithm is not directly applicable when the image data set is correlated in higher dimensions (namely S^2 and $SO(3)$). This results from the fact that the right singular vectors of these data sets are no longer approximately spanned by the first few Fourier harmonics. It will be shown however, that the spectral information of these image data sets can still be efficiently computed by considering the SHT in place of the FFT [27–29]. Spherical harmonics and the SHT algorithm on S^2 are presented in the next chapter.

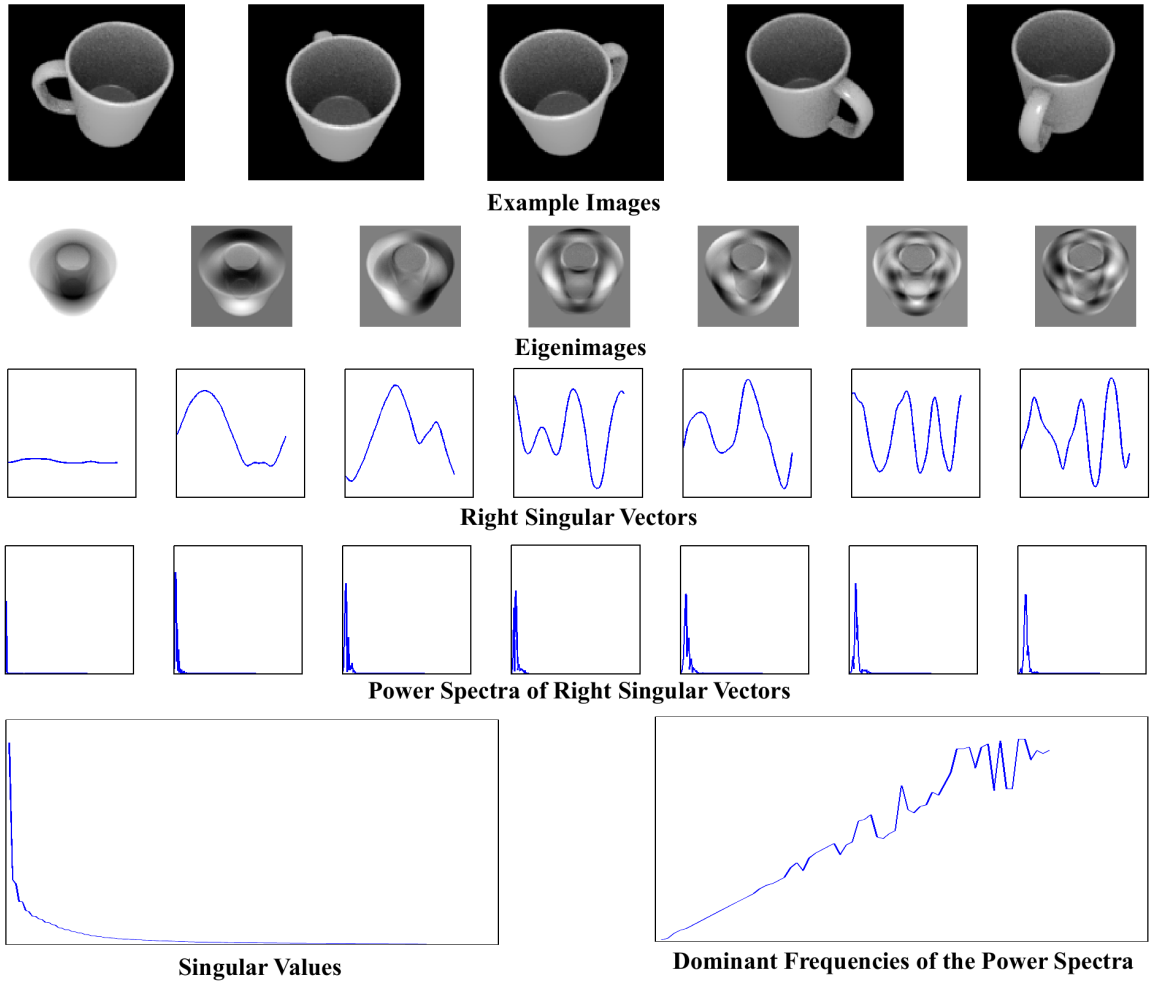

Figure 3: The eigenspace decomposition of the image matrix X obtained from sampling a coffee mug according to the method described for S^1 correlation. The first row shows five of the 128 images of the image data matrix X. The second row shows the first seven eigenimages (left singular vectors of X). The third row shows the first seven right singular vectors of X. The fourth row shows the power spectra of these right singular vectors. It is apparent that though the right singular vectors of X are not pure sinusoids, their power spectra are concentrated in a narrow band around frequencies that are harmonics of $2\pi/n$. The plot on the left in the last row shows the singular values of X, while the plot on the right shows the frequency at which the power spectra of the corresponding right singular vectors achieves a maximum (i.e., the "dominant" frequencies). It can be seen that the dominant frequencies of the power spectra of the right singular vectors corresponding to nonzero singular values increase approximately linearly with their index.

CHAPTER IV

SPHERICAL TESSELLATION ANALYSIS

4.1 Chapter Overview

Prior to the development of a discrete SHT, the best tessellation of the sphere to define the sampling pattern needs to be determined. This chapter outlines three tessellations that are commonly used to discretize the sphere. Computing the SHT of a function defined on the sphere will be developed in Chapter 5, however for the discussion at hand, a generalization is outlined in Section 4.2. The three tessellations outlined are then presented in Sections 4.3, 4.4, and 4.5, with an analysis of all three presented in Section 4.6.

4.2 Introduction

Given a function $f_{i,j}$ discretely sampled on the sphere, where the latitudes are defined by $\theta_i,\ i = 0, \ldots, N$ and longitudes are defined by $\phi_j,\ j = 0, \ldots, 2N-2$, the function can be represented by the forward harmonic transform as

$$f_{i,j} = \sum_{n=0}^{N-1} \sum_{m=0}^{n} P_n^m(\theta_i)(a_{m,n}\cos(m\phi_j) + b_{m,n}\sin(m\phi_j), \tag{42}$$

where $P_n^m(\cdot)$ is an associated Legendre polynomial. Notice that the co-latitudinal coordinate does not depend on m, therefore, for each θ_i, the coefficients for the longitudinal coordinate are given by

$$\begin{aligned} a_m(\theta_i) &= \tfrac{1}{2N-2} \sum_{j=1}^{2N-2} f_{i,j}\cos(m\phi_j) \\ b_m(\theta_i) &= \tfrac{1}{2N-2} \sum_{j=1}^{2N-2} f_{i,j}\sin(m\phi_j). \end{aligned} \tag{43}$$

The coefficients for the co-latitudinal coordinate can be then computed as

$$\begin{aligned} a_{m,n} &= \sum_{i=1}^{N} w_i a_m(\theta_i) P_n^m(\theta_i) \\ b_{m,n} &= \sum_{i=1}^{N} w_i b_m(\theta_i) P_n^m(\theta_i) \end{aligned} \tag{44}$$

where the w_i are quadrature weights that are dependent on the particular tessellation being employed. Three popular tessellations continually arise in the literature when discussing spectral analysis on the surface of the sphere, each of which will be discussed in the following sections [46].

4.3 Gauss-Legendre Grid

In [47,48], Swarztrauber *et al.* proposed a method for computing the discrete SHT using the Gauss-Legendre grid, as well as an efficient method for computing the quadrature weights and points [49]. If the samples are defined by a Gaussian distribution in the co-latitudinal coordinate, then the harmonic coefficients $a_{m,n}$ and $b_{m,n}$ defined in (44) can be developed as an application of the matrix operator $\mathbf{P}_m^T\mathbf{W}$ where $\mathbf{W}$ is the $N \times N$ diagonal matrix of Gaussian weights w_i and $\mathbf{P}_m$ is an $N \times (N-m)$ matrix

$$\mathbf{P}_m = \begin{bmatrix} P_m^m(\theta_1) & \cdots & P_{N-1}^m(\theta_1) \\ \vdots & \ddots & \vdots \\ P_m^m(\theta_N) & \cdots & P_{N-1}^m(\theta_N) \end{bmatrix} \tag{45}$$

where $P_{\dot{()}}^{\dot{()}}(\theta_i)$ are the associated Legendre functions [48]. The forward transform defined in (42), can be computed by applying the matrix $\mathbf{P}_m$ to the harmonic coefficients defined in (44). The complete Legendre projection that encapsulates both the forward and inverse harmonic transform is then defined as

$$\mathbf{F}_m = \mathbf{P}_m \mathbf{P}_m^T \mathbf{W}. \tag{46}$$

Because $\mathbf{F}_m$ is a projection, $\mathbf{F}_m^2 = \mathbf{F}_m = \mathbf{P}_m\mathbf{P}_m^T\mathbf{W}\mathbf{P}_m\mathbf{P}_m^T\mathbf{W}$, which implies that $\mathbf{P}_m^T\mathbf{W}\mathbf{P}_m = I_{(N-m)\times(N-m)}$ [48]. Letting $m = 0$,

$$\begin{aligned} &\mathbf{P}_0^T\mathbf{W}\mathbf{P}_0 = I_{N\times N} \\ \implies &\mathbf{P}_0\mathbf{P}_0^T\mathbf{W}\mathbf{P}_0 = \mathbf{P}_0 \\ \implies &\mathbf{P}_0\mathbf{P}_0^T\mathbf{W}\mathbf{P}_0(\mathbf{P}_0)^{-1} = \mathbf{P}_0(\mathbf{P}_0)^{-1} \\ \implies &\mathbf{P}_0\mathbf{P}_0^T\mathbf{W} = I_{N\times N}, \end{aligned} \tag{47}$$

and the weights can be computed as

$$\mathbf{W} = (\mathbf{P}_0\mathbf{P}_0^T)^{-1}. \tag{48}$$

To compute the points, consider the elements of $\mathbf{P}_0\mathbf{P}_0^T$ for an arbitrary distribution of co-latitudes θ_i,

$$(\mathbf{P}_0\mathbf{P}_0^T)_{i,j} = \sum_{k=0}^{N-1} P_k^0(\theta_i)P_k^0(\theta_j). \tag{49}$$

It is shown in [49] that $\mathbf{P}_0\mathbf{P}_0^T$ can be written as

$$(\mathbf{P}_0\mathbf{P}_0^T)_{i,j} = \frac{N}{\sqrt{4N^2-1}}\frac{P_N^0(\theta_i)P_{N-1}^0(\theta_j) - P_{N-1}^0(\theta_i)P_N^0(\theta_j)}{\sin(\theta_i) - \sin(\theta_j)}, \tag{50}$$

with the diagonal entries computed as

$$(\mathbf{P}_0\mathbf{P}_0^T)_{i,i} = \frac{N}{\sqrt{4N^2-1}\cos(\theta_i)}\left[P_{N-1}^0(\theta_i)\frac{d}{d\theta}P_N^0(\theta_i) - P_N^0(\theta_i)\frac{d}{d\theta}P_{N-1}^0(\theta_i)\right]. \tag{51}$$

If the θ_i are chosen to be the zeros of $P_N^0(\theta_i)$, then the resulting matrix is diagonal and the θ_i are Gaussian distributed. The corresponding quadrature weights have the closed form solution

$$(\mathbf{W})_{i,i} = \frac{\sqrt{4N^2-1}\cos(\theta_i)}{NP_{N-1}^0(\theta_i)\frac{d}{d\theta}P_N^0(\theta_i)}. \tag{52}$$

Using the Gaussian points to define the sampling pattern on the sphere allows for an exact quadrature approximation to the transform defined in (44) when the corresponding Gaussian weights are used. An example of the Gauss-Legendre sampling pattern is depicted in Fig. 4. In the figure, a sample image is taken at each of the black dots on the surface of the sphere.

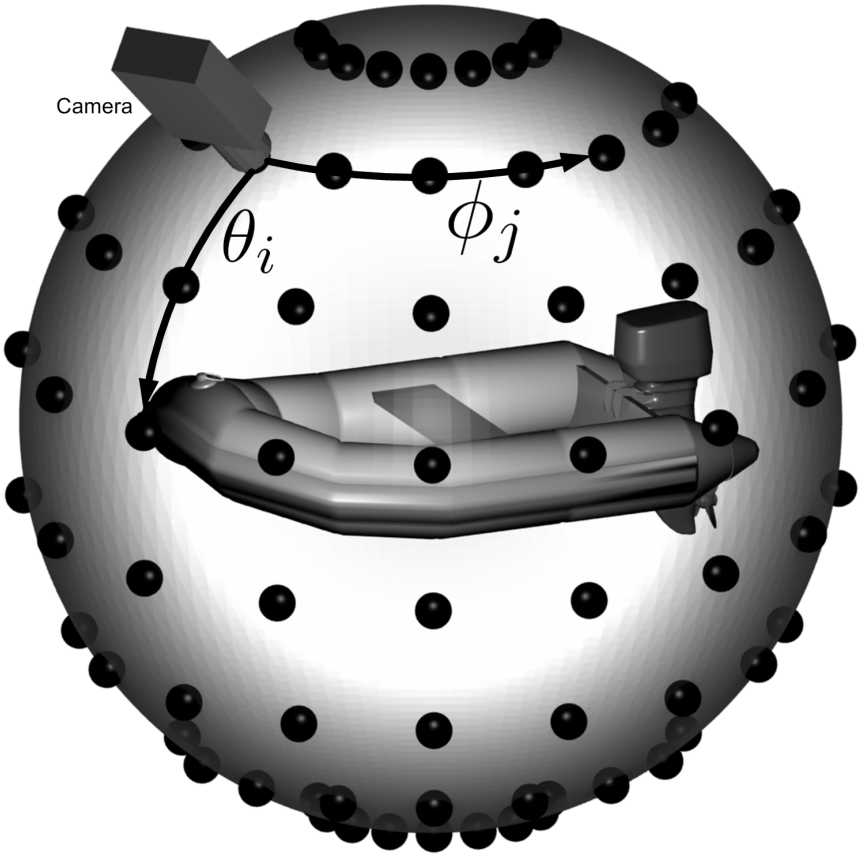

Figure 4: The Gauss-Legendre grid of sample points defined on the sphere. A sample image is taken at each of the black dots on the sphere.

4.4 Equi-Angular Grid of Chebychev Nodes

An alternate method to the Gauss-Legendre points and weights is to use an equi-angular grid of Chebychev nodes [50,51]. Driscoll and Healy formulate the problem as follows: Given a band-limited function with band limit B whose domain is restricted to the sphere, define the sampling of this function as $\theta_i = \frac{\pi(2i+1)}{4B}$ as the co-latitudinal component, and $\phi_j = \frac{2\pi j}{2B}$ as the longitudinal component with $0 < i, j < 2B$ being integers. This sampling pattern defines an equi-angular grid of Chebychev nodes on the surface of the sphere. The forward harmonic transform defined in (44) can then be computed with the weights w_i defined as the solution to the system of linear equations

$$\sum_{i=0}^{2B-1} w_i P_m(\cos\theta_i) \tag{53}$$

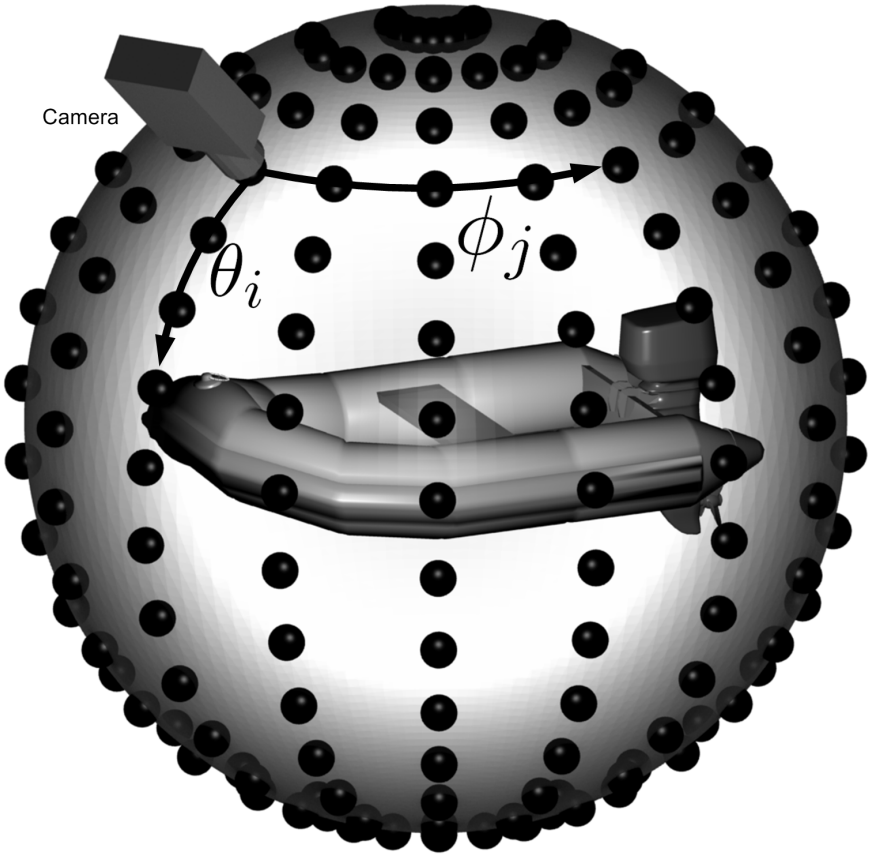

Figure 5: The equi-angular grid of Chebychev points defined on the sphere. A sample image is taken at each of the black dots on the sphere.

where $P_m(\cdot)$ is the Legendre function of order m. The solution to (53) can be computed in closed form as [50]

$$w_i = \frac{2}{B}\sin\left(\frac{\pi(2i+1)}{4B}\right)\sum_{k=0}^{B-1}\sin\left((2i+1)(2k+1)\frac{\pi}{4B}\right). \tag{54}$$

Using the above weights and points, the equi-angular grid also enjoys the benefit of being weighted orthogonal in the discrete Legendre transform. That is, $\mathbf{P}_m^T\mathbf{W}\mathbf{P}_m = I_{(2B-m)\times(2B-m)}$, where $\mathbf{P}_m$ is defined in (45) and $\mathbf{W}_{i,i} = w_i$ is a diagonal matrix. As in the case of the Gauss-Legendre grid, the quadrature approximation to the transform defined in (44) is exact when using the equi-angular grid and corresponding weights defined in (54). An example of the equi-angular sampling pattern is depicted in Fig. 5. In the figure, a sample image is taken at each of the black dots on the surface of the sphere.

4.5 HEALPix Grid

In [52], Górski *et al.* propose the Hierarchical Equal Area isoLatitude Pixelization (HEALPix) to define the sampling over the sphere. While the above two tessellations have either closed form representations for computing the sample points (equi-angular) or can be computed by solving for the roots of an m^{th} order Legendre polynomial (Gauss-Legendre), the HEALPix tessellation is fundamentally different. Originally developed for fast analysis of data generated from the cosmic microwave background (CMB) experiments, the HEALPix tessellation is based on sub-dividing a base tessellation using the parameter N_{side}.

The construction of the HEALPix sampling pattern is as follows [52]: N_θ is the number of base resolution sample layers between the north and south poles, and N_ϕ is the multiplicity of the equatorial base resolution samples. The total number of base resolution samples is then given by the product of the two $N_{\text{base}} = N_\theta N_\phi$, with the area of each base resolution sample given by $\Omega_{\text{base}} = 4\pi/(N_\theta N_\phi)$ (an example of the base resolution samples used in this study is shown in Fig. 6). By construction, the tessellation includes two layers of polar cap samples, and $N_\theta - 2$ layers of equatorial zone samples that form a rhomboidal grid in the cylindrical projection of the sphere. Because the cylindrical projection on the sphere is an area preserving operator, all rhomboidal sampling zones in the equatorial region have equal area weighting. It is desired to have equal area weighting in sample points distributed over the entire sphere, therefore, constraining the co-latitudinal component to $\cos(\theta) = (N_\theta - 1)/N_\theta$, the polar samples zones become quadrilaterals that vary in shape but retain equal area. Therefore, the HEALPix tessellation has the advantage of equal area weighting in sampling the sphere.

The authors of [52] chose $N_\theta = 3$ and $N_\phi = 4$ (refer to Fig. 6) for the derivation of the HEALPix grid for three reasons:

1. No more than four samples are at the polar cap.
2. The elongation of the equatorial rhomboidal regions is minimized
3. The 2^n multiplicity of rings in the equatorial zone is retained for the fast harmonic transform.

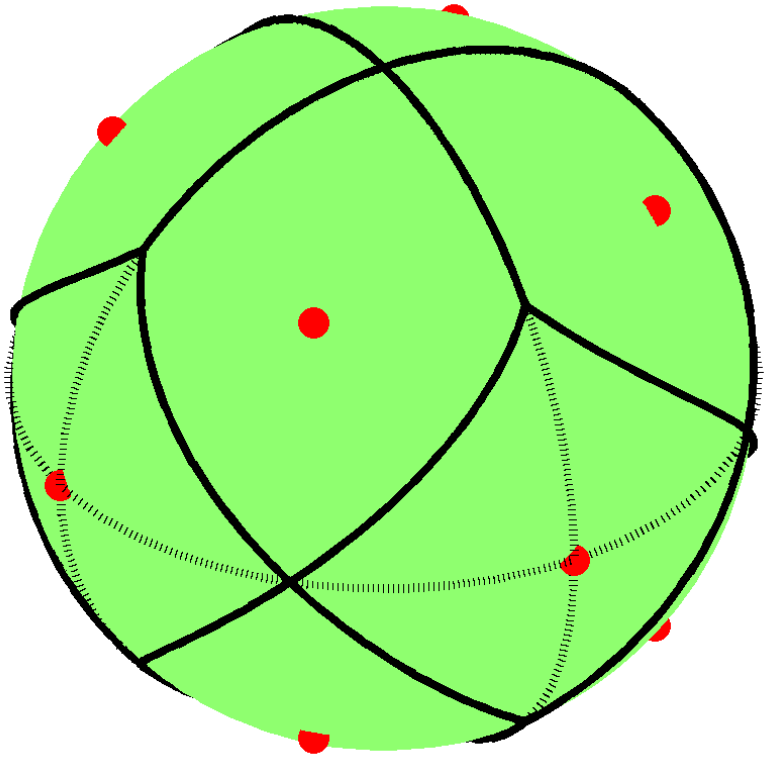

Figure 6: The base resolution layers for the HEALPix tessellation using the parameters $N_\theta = 3$ and $N_\phi = 4$.

Because of the above three reasons, the HEALPix tessellation provides a very good sampling method for the problem of pose estimation as well. The first item shows that the polar regions are not over sampled as they are in the two previous tessellations. This has a direct effect on the SVD in that the polar images have higher weighting and the SVD is biased toward them. The second item has no direct effect on the pose estimation problem, however it provides a sampling pattern in the equatorial region that is evenly distributed (giving equal weight to the images captured around the equator as well). The third item is directly related to keeping the SHT as fast as possible in the longitudinal direction. The sample points for the HEALPix tessellation can be computed as follows [52]:

First compute the sample points for the *North polar cap*:

$$p = 0,\ p_h = (p+1)/2, \text{ and } i = \lfloor \sqrt{p_h - \sqrt{\lfloor p_h \rfloor}} \rfloor + 1$$

while$((i \geq 1)$ and $(i \leq N_{\text{side}}))$

$j = p + 1 - 2i(i-1)$

while$((j \geq 1)$ and $(j \leq 4i))$

$i = \lfloor \sqrt{p_h - \sqrt{\lfloor p_h \rfloor}} \rfloor + 1$

$j = p + 1 - 2i(i-1)$

$z = 1 - \frac{i^2}{3N_{\text{side}}^2}$

$\theta = \text{acos}(z)$

$\phi = \pi/(2i)(j + 0.5)$

$p = p + 1$

$p_h = (p+1)/2$

end while

$i = \lfloor \sqrt{p_h - \sqrt{\lfloor p_h \rfloor}} \rfloor + 1$

end while

Next compute the sample points for the *North equatorial belt*:

$p' = p - 2N_{\text{side}}(N_{\text{side}} - 1)$

$i = \lfloor p'/(4N_{\text{side}}) \rfloor + N_{\text{side}}$

while$((i \geq 1)$ and $(i \leq 2N_{\text{side}}))$

$j = \text{mod}(p', 4N_{\text{side}}) + 1$

while$((j \geq 1)$ and $(j < 4N_{\text{side}}))$

$i = \lfloor p'/(4N_{\text{side}}) \rfloor + N_{\text{side}}$

$j = \text{mod}(p', 4N_{\text{side}}) + 1$

$z = \frac{4}{3} - \frac{2i}{3N_{\text{side}}}$

$\theta = \text{acos}(z)$

$s = \text{mod}((i - N_{\text{side}} + 1), 2)$

$\phi = \pi/(2N_{\text{side}})(j - s/2)$

$p = p + 1$

$$p' = p - 2N_{\text{side}}(N_{\text{side}} - 1)$$

end while

$$i = \lfloor p'/(4N_{\text{side}}) \rfloor + N_{\text{side}}$$

end while

The sample points in the southern hemisphere are obtained by the mirror symmetry about the equator of the sphere. Note that in the above algorithm, $\lfloor x \rfloor$ is the highest integer less than or equal to x.

While the HEALPix sampling pattern provides equal area weighting over the surface of the sphere, there is no exact quadrature to make the transform defined in (44) exact. A numerical quadrature can be calculated by iterative methods, however this is computationally expensive and needs to be carried out for every order of the Legendre functions. As a result, numerical errors may exist in the harmonic transform as defined in (44). The accumulation of error continues to increase as the bandwidth of the function increases. For pose estimation problems however, relatively low bandwidths are needed and the numerical error is minimal. An example of the HEALPix tessellation as it applies to pose estimation is shown in Fig. 7. In the figure, as above, a sample image is taken at each of the black dots on the surface of the sphere.

4.6 Tessellation Evaluation

In an attempt to determine which of the above tessellations performs the best for this particular application, the following test was performed. First, several objects were sampled using the three above mentioned tessellations (examples of which are shown in Fig. 8). Next, the spherical harmonic transform on S^2 described in (44) was computed for each of these data sets and 256 of the harmonic images with the largest power spectra were kept (refer to Table 1). Finally, the SVD was performed on the resulting harmonic images to give the approximate eigenspace. The true eigenspace was also computed by using the direct SVD of the resulting image data.[1]

[1]Note that the image data matrices are different for each of the different tessellations.

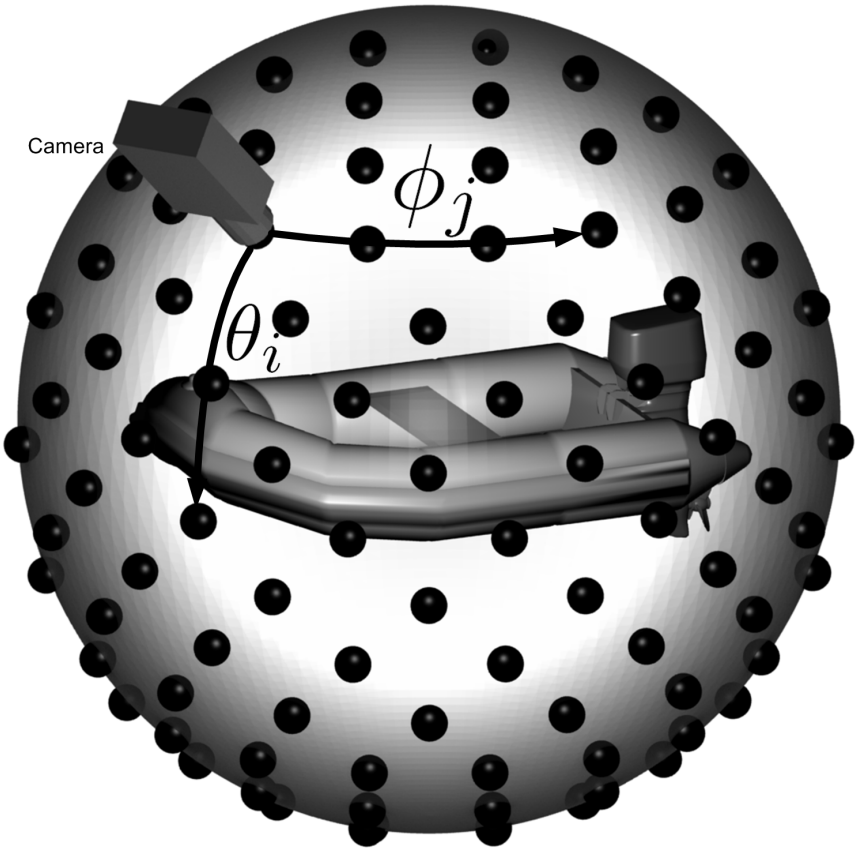

Figure 7: The HEALPix grid of sample points defined on the sphere. A sample image is taken at each of the black dots on the sphere.

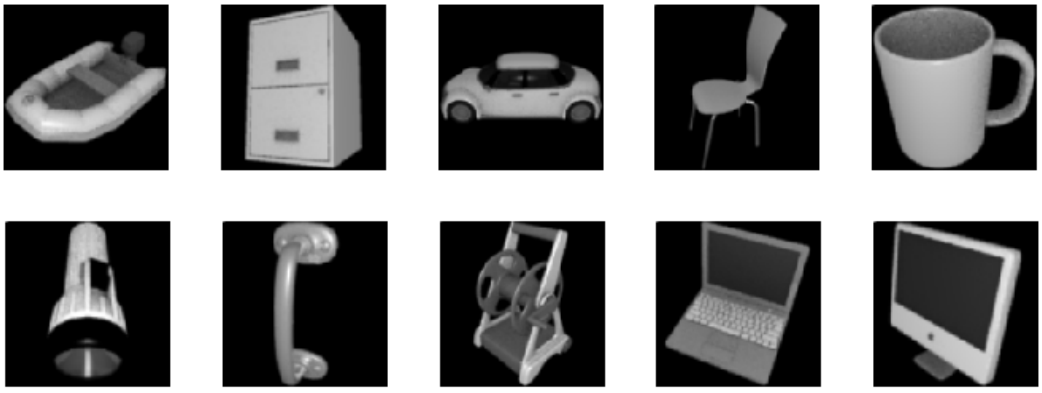

Figure 8: A sample image of the ten different objects that were used in this analysis. Each of the objects was sampled according to one of the three tessellations mentioned above. The images were generated by ray-tracing CAD models for each object. The CAD models were provided courtesy of Kator Legaz [1]

Table 1: Sample information for each of the three tessellations discussed in this chapter.

	Gauss-Legendre	Equi-angular	HEALPix
# Samples	512	1024	768
# Harmonics	256	256	256

Fig. 9 shows the difference between the subspace dimension as computed by the SVD(X) and all three tessellations for an energy recovery ratio of $\mu = 0.9$. As can be seen from the figure, the HEALPix sampling grid gives a very good estimate of the subspace dimension as compared to the true SVD. This estimate is extremely important in that the on-line computation time is dependent on this being as small as possible. As mentioned in Chapter 2, the true SVD gives the optimal subspace; therefore, the quality of the estimated subspace is also important. The quality measures outlined in Chapter 2 were computed for all three tessellations and averaged over all objects in Fig. 8. The results are depicted in Fig. 10. As can be seen from the figure, all three tessellations provide a very good estimate of the true eigenspace decomposition, with the HEALPix tessellation giving the best performance.

Even though the HEALPix tessellation has no exact quadrature, based on the above analysis, the errors associated with this are minimal. It has been shown in [53] that for arbitrary functions defined on S^2, the performance and computation times in computing

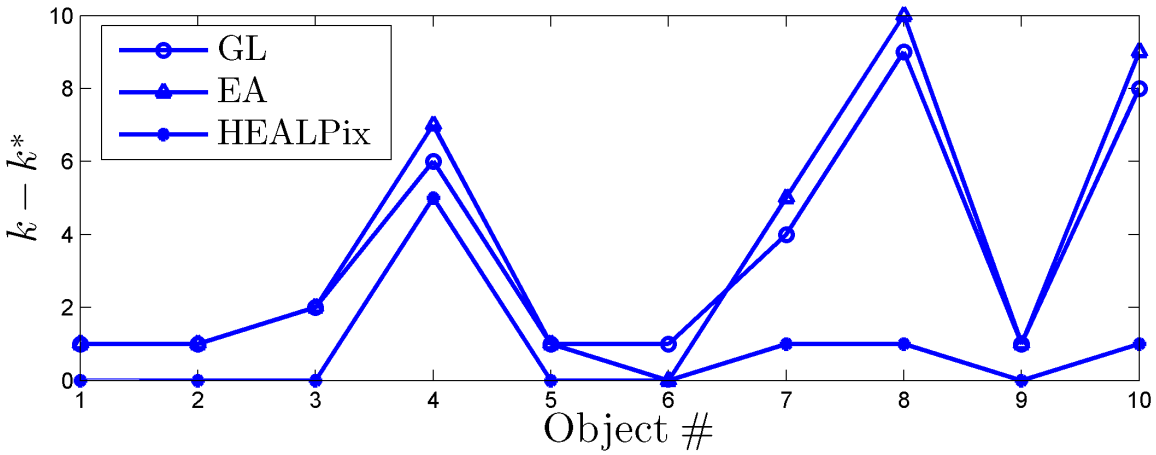

Figure 9: Difference between the subspace dimension computed by the SVD and that computed for each of the three tessellations. The subspace dimension computed is for a recovery of over 90% of the energy in the original image data matrix. In the above figure, GL represents the Gauss-Legendre grid and EA represents the equi-angular grid.

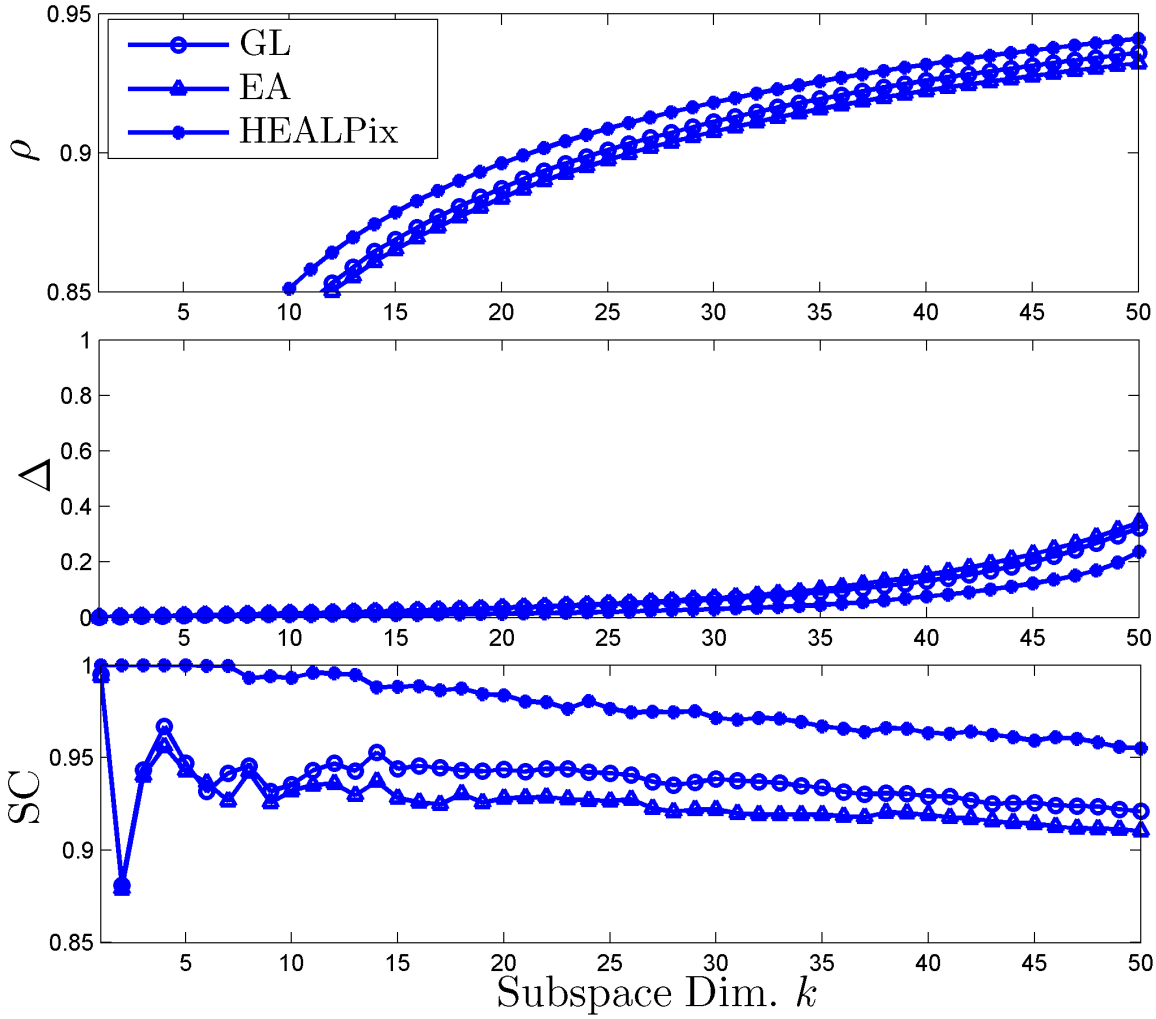

Figure 10: Quality measures outlined in Chapter 2 computed for all three tessellations and averaged over all test objects. As can be seen from the figure, the HEALPix tessellation outperforms the other two in terms of its capability of representing the "true" eigenspace as computed by the SVD. In the above figure, GL represents the Gauss-Legendre grid and EA represents the equi-angular grid.

both the forward and inverse transform using the HEALPix tessellation is nearly equivalent to using the equi-angular tessellation. However, this is a function of the application and the results presented here suggest that for the pose estimation problem, the HEALPix tessellation outperforms the other two. This result is based on the fact that for a similar number of samples, the HEALPix tessellation gives better angular resolution and does not bias the SVD by oversampling the polar regions. Based on this analysis, the HEALPix sampling pattern is used to define the sampling over S^2 for the remainder of this work.

CHAPTER V

SPHERICAL HARMONICS AND THE 2-SPHERE

5.1 Chapter Overview

This chapter gives an overview of harmonic analysis on S^2. Before developing this analysis however, harmonic analysis on S^1 is developed in Section 5.3, which leads to the standard Fourier basis. This development is then extended to S^2 in Section 5.4 by replacing the Fourier basis with the standard spherical harmonics leading to the development of the spherical harmonic transform outlined in Section 5.5. This development is then used to construct a computationally efficient algorithm for computing the eigenspace decomposition of image data sets correlated on S^2, detailed in Section 5.6. In Section 5.7, experimental results are presented validating the computational savings and accuracy of estimation, as well as providing a maximum recoverable energy using the proposed method. Finally, the chapter is concluded by showing how pose estimation from an aerial perspective can be achieved by sampling the upper hemisphere of S^2 and replacing the spherical harmonic basis with a set of hemispherical harmonics.

5.2 Introduction

Spherical harmonics have been applied to a variety of problems that arise on the surface of the unit sphere (denoted as the 2-sphere or S^2). They have been used for solving PDE's in spherical geometry for weather and climate models [54], geophysics [55, 56], quantum mechanics [57,58], as well as a host of other related applications [59]. Over the last decade, spherical harmonics have been gaining popularity in the computer vision and computer graphics arena. Spherical harmonics have been applied to several computer vision applications with unknown lighting [30, 60–63], as well as 3-D model retrieval [64, 65], and 3-D

shape descriptors [66]. Spherical harmonics have also been applied to rotation estimation and convolution of spherical images [67].

5.3 Harmonic Analysis on S^1

Before developing the harmonic analysis on $S^2 \in \mathbb{R}^3$, we first develop the harmonic analysis on $S^1 \in \mathbb{R}^2$ leading to a standard Fourier series, much of which is discussed in [68–70]. The development of harmonic analysis on S^1 shows that the Fourier series basis vectors have a natural extension to spherical harmonics using the Laplacian operator ∇^2. For this development, our attention is restricted to harmonic homogeneous polynomials [69].

A harmonic homogeneous polynomial $p_n(x, y)$ of degree n is one that satisfies $\nabla_2^2 p_n(x, y) = 0$ (harmonic), and $p_n(tx, ty) = t^n p_n(x, y)$ where $t > 0$ (homogeneous). The Laplacian operator $\nabla_2^2 \in \mathbb{R}^2$ is defined as

$$\nabla_2^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}, \tag{55}$$

where the subscript denotes the dimension of the Laplacian.

The representation of a polynomial in spherical coordinates $\{(r, \alpha) \ : \ r \in \mathbb{R}_+, \alpha \in [0, 2\pi)\}$ can be defined as $(x = r\cos(\alpha), \ y = r\sin(\alpha))$. Using spherical coordinates, a harmonic homogeneous polynomial $p_n(x, y)$ can be written as $p_n(x, y) = r^n q_n(\alpha)$ for some function $q_n(\alpha)$. The two-dimensional Laplacian defined in spherical coordinates is

$$\nabla_2^2 = \frac{1}{r^2}\frac{\partial^2}{\partial \alpha^2} + \frac{\partial^2}{\partial r^2} + \frac{1}{r}\frac{\partial}{\partial r}. \tag{56}$$

If the polynomial $p_n(x, y)$ is homogeneous and harmonic, then it must satisfy

$$\nabla_2^2 p_n(x, y) = r^{n-1}\left(\frac{\partial^2}{\partial \alpha^2} q_n(\alpha) + n^2 q_n(\alpha)\right) = 0 \tag{57}$$

where $\nabla_{2S}^2 = \frac{\partial^2}{\partial \alpha^2}$ is the circular Laplacian. This shows that the Laplacian operator applied to a harmonic homogeneous polynomial simply reduces the power by two in the radial dimension, and restricts the angular dimension to a circle using the circular Laplacian. This allows us to separate variables and consider the radial dimension and angular dimension separately in solving for $q_n(\alpha)$. In the radial dimension r, there is no $q_n(\alpha)$ dependence, however, in the angular dimension, we have that $\nabla_{2S}^2 q_n(\alpha) = -n^2 q_n(\alpha)$. In order for (57) to

hold, $q_n(\alpha)$ must be an eigenfunction of ∇^2_{2S}. The $q_n(\alpha)$ that satisfy the differential equation $\nabla^2_{2S} q_n(\alpha) = -n^2 q_n(\alpha)$ are of the form $q_n(\alpha) = A\cos(n\alpha)+B\sin(n\alpha)$ or equivalently, $q_n(\alpha) = Ae^{in\alpha} + Be^{-in\alpha}$. Therefore, all possible $q_n(\alpha)$ can be generated as a linear combination of the elements $\{e^{in\alpha}, e^{-in\alpha}\}$. If the exponentials $\{e^{in\alpha}, e^{-in\alpha}\}$ are normalized and chosen to be orthogonal, then they form the standard Fourier basis vectors.

5.4 Harmonic Analysis on S^2

Extending the analysis of the previous section to S^2 is a matter of increasing the dimension by one. Again, we restrict our attention to harmonic homogeneous polynomials, however as in the case for S^1, this restriction has no direct implications [69]. In $\mathbb{R}^3$, we define the three dimensional Laplacian as

$$\nabla^2_3 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \tag{58}$$

and as above, a harmonic homogeneous polynomial $p_n(x, y, z)$ of degree n in $\mathbb{R}^3$ is one that satisfies $\nabla^2_3 p_n(x, y, z) = 0$, and $p_n(tx, ty, tz) = t^n p_n(x, y, z)$ where $t > 0$.

Using spherical coordinates, $\{(r, \alpha, \beta) \ : \ r \in \mathbb{R}_+, \alpha \in [0, 2\pi), \beta \in [0, \pi]\}$ the cartesian coordinates can be defined as $(x = r\cos(\alpha)\sin(\beta),\ y = r\sin(\alpha)\sin(\beta)\ z = r\cos(\beta))$. Under this coordinate frame, the Laplacian becomes

$$\nabla^2_3 = \frac{\partial}{\partial r^2} + \frac{2}{r}\frac{\partial}{\partial r} + \frac{1}{r^2\sin^2(\beta)}\frac{\partial^2}{\partial \alpha^2} + \frac{\cos(\beta)}{r^2\sin(\beta)}\frac{\partial}{\partial \beta} + \frac{1}{r^2}\frac{\partial^2}{\partial \beta^2}, \tag{59}$$

and again we have that $p_n(x, y, z)$ can be written as $p_n(x, y, z) = r^n q_n(\alpha, \beta)$ for some $q_n(\alpha, \beta)$. Again, if p_n is homogeneous and harmonic, it must satisfy

$$\nabla^2_3 p_n = \left[\csc^2(\beta)\frac{\partial^2}{\partial \alpha^2} + \frac{\partial^2}{\partial \beta^2} + \cot(\beta)\frac{\partial}{\partial \beta}\right] q_n(\alpha, \beta) + n(n+1) q_n(\alpha, \beta) = 0 \tag{60}$$

where the spherical Laplacian is now defined by

$$\nabla^2_{3S} = \left[\csc^2(\beta)\frac{\partial^2}{\partial \alpha^2} + \frac{\partial^2}{\partial \beta^2} + \cot(\beta)\frac{\partial}{\partial \beta}\right]. \tag{61}$$

Using this spherical Laplacian, we have that $\nabla^2_{3S} q_n(\alpha, \beta) = -n(n+1) q_n(\alpha, \beta)$ and therefore, $q_n(\alpha, \beta)$ is an eigenfunction of ∇^2_{3S}. Because the spherical Laplacian is self-adjoint (the circular Laplacian defined above is also), the eigenspaces of ∇^2_{3S} are orthogonal and rotation

invariant. These eigenspaces consist of harmonic polynomials of degree $l = n$ and have dimension $2l + 1$. On the surface of the sphere, these harmonic polynomials span the set of all polynomials in L^2, where L^2 is the Hilbert space of square integrable functions. Therefore, if we select a basis within each of the $2l + 1$ eigenspaces, this collection provides a basis for $L^2(S^2)$.

For fixed l, the eigenspaces can be constructed as

$$\frac{1}{\sqrt{2\pi}}\tilde{P}_l^m(\cos(\beta))e^{im\alpha} \tag{62}$$

where the exponential term is a direct result of the longitudinal coordinate α defining circles at constant co-latitude β, i.e., the basis for the longitudinal coordinate is the standard Fourier series developed above. Using this construction, we have

$$\begin{aligned} \nabla_{3S}^2 \frac{e^{im\alpha}}{\sqrt{2\pi}}\tilde{P}_l^m(\cos(\beta)) &= \left[\frac{\partial^2}{\partial\beta^2} + \cot(\beta)\frac{\partial}{\partial\beta} - \frac{m^2}{\sin^2(\alpha)}\right]\frac{e^{im\alpha}}{\sqrt{2\pi}}\tilde{P}_l^m(\cos(\beta)) \\ &= -l(l+1)\frac{e^{im\alpha}}{\sqrt{2\pi}}\tilde{P}_l^m(\cos(\beta)), \end{aligned} \tag{63}$$

where the $-m^2$ comes from explicit differentiation in the longitudinal coordinate. Under the condition that $\beta \in (0, \pi)$, the functions $\tilde{P}_l^m(\cos(\beta))$ are the associated Legendre polynomials of order m and degree l. The tilde ($\tilde{\cdot}$) denotes the L^2 normalized versions of the associated Legendre polynomials and are related to their un-normalized cousins by

$$\tilde{P}_l^m(\cos(\beta)) = (-1)^m\sqrt{\frac{2l+1}{2}\frac{(l-m)!}{(l+m)!}}P_l^m(\cos(\beta)). \tag{64}$$

The associated Legendre polynomials may be computed efficiently using the following three term recurrence relationship

$$(l-m+1)P_{l+1}^m(x) = (2l+1)xP_l^m(x) - (l+m)P_{l-1}^m(x), \tag{65}$$

where $x = \cos(\beta)$. Note that the functions $\tilde{P}_l^m(\cos(\beta))$ are orthonormal with respect to the latitudinal angle β and the functions $\frac{e^{im\alpha}}{\sqrt{2\pi}}$ are orthonormal with respect to the longitudinal angle α. Taking the product of the two we get the classical spherical harmonics

$$Y_l^m(\alpha,\beta) = (-1)^m\sqrt{\frac{2l+1}{4\pi}\frac{(l-m)!}{(l+m)!}}P_l^m(\cos(\beta))e^{im\alpha} \tag{66}$$

that satisfy

$$\int_0^{2\pi}\int_0^{\pi} Y_{l_1}^{m_1}(Y_{l_2}^{m_2})^*\sin(\beta)d\beta d\alpha = \delta_{l_1 l_2}\delta_{m_1 m_2}, \tag{67}$$

where * denotes complex conjugation, i.e., they form an orthonormal basis over the surface of the sphere [58].

The spherical harmonics form a triangular truncation with $|m| \leq l$, and $l = 0, 1, \ldots, \infty$. This triangular truncation is shown in Fig. 11, that depicts the spherical harmonics for degrees $l = 0, 1, 2$. Note that the spherical harmonics for negative m are simply phase-shifted versions of their positive m counterparts. The analytic expressions for the first few spherical harmonics are given in Table 2.

The spherical harmonics can also be categorized depending on the order m. When $m = 0$, their is no longitudinal contribution because the exponentials $e^{im\alpha} = 1$. Harmonics evaluated when $m = 0$ are referred to as the zonal harmonics and simply determine the harmonic contribution of the function in the co-latitudinal coordinate. Examples of the zonal harmonics projected onto the sphere, along with their associated Legendre polynomials are shown in Fig. 12. If on the other hand $m = l$, then the associated Legendre polynomials

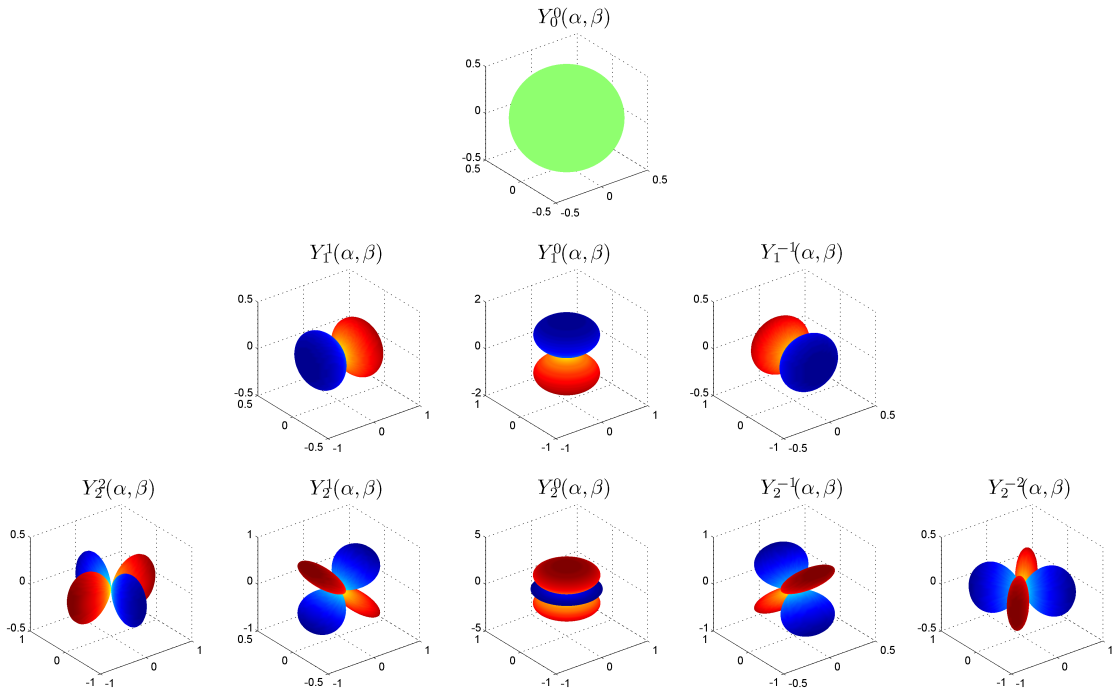

Figure 11: The first few spherical harmonics showing the triangular truncation. For negative values of m, the harmonics are simply phase-shifted versions of their positive m counterparts.

Table 2: Analytic expressions for spherical harmonics of degree $l = 0, 1, 2$ and $|m| \leq l$.

		m				
		-2	-1	0	1	2
l	0	NA	NA	$\frac{1}{2}\sqrt{\frac{1}{\pi}}$	NA	NA
	1	NA	$\frac{1}{2}\sqrt{\frac{3}{2\pi}}\sin(\beta)e^{-i\alpha}$	$\frac{1}{2}\sqrt{\frac{3}{2\pi}}\cos(\beta)$	$-\frac{1}{2}\sqrt{\frac{3}{2\pi}}\sin(\beta)e^{i\alpha}$	NA
	2	$\frac{1}{4}\sqrt{\frac{15}{2\pi}}\sin^2(\beta)e^{-2i\alpha}$	$\frac{1}{2}\sin(\beta)\cos(\beta)e^{-i\alpha}$	$\frac{1}{4}\sqrt{\frac{5}{\pi}}\left(3\cos^2(\beta)-1\right)$	$-\frac{1}{2}\sin(\beta)\cos(\beta)e^{i\alpha}$	$\frac{1}{4}\sqrt{\frac{15}{2\pi}}\sin^2(\beta)e^{2i\alpha}$

simply attenuate the polar regions and the harmonics are referred to as the sectoral harmonics. The sectoral harmonics determine the harmonic contribution of the function in the longitudinal coordinate. Examples of the sectoral harmonics, along with their associated Legendre polynomials are shown in Fig. 13. Finally, when $m \neq (0, l)$, the harmonics are referred to as the tesseral harmonics and they determine the harmonic contribution of the function in the cross combination of the co-latitudinal and longitudinal coordinate. Examples of the tesseral harmonics, along with their associated Legendre polynomials are shown in Fig. 14.

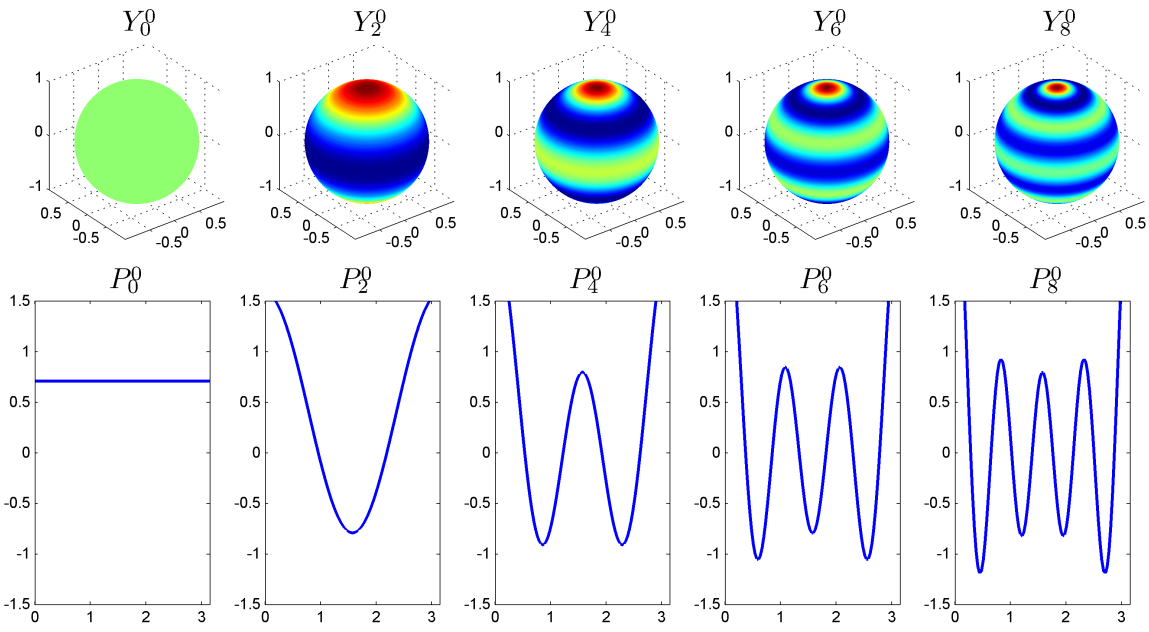

Figure 12: In the definition of the spherical harmonics $Y_l^m(\cdot)$, when $m = 0$ there is no longitudinal dependence. As a result, only the harmonic frequencies in the co-latitudinal coordinate are computed. These are referred to as the zonal harmonics shown in the top figures for several different harmonic degrees l. The bottom figures shows the contribution of the Legendre polynomials.

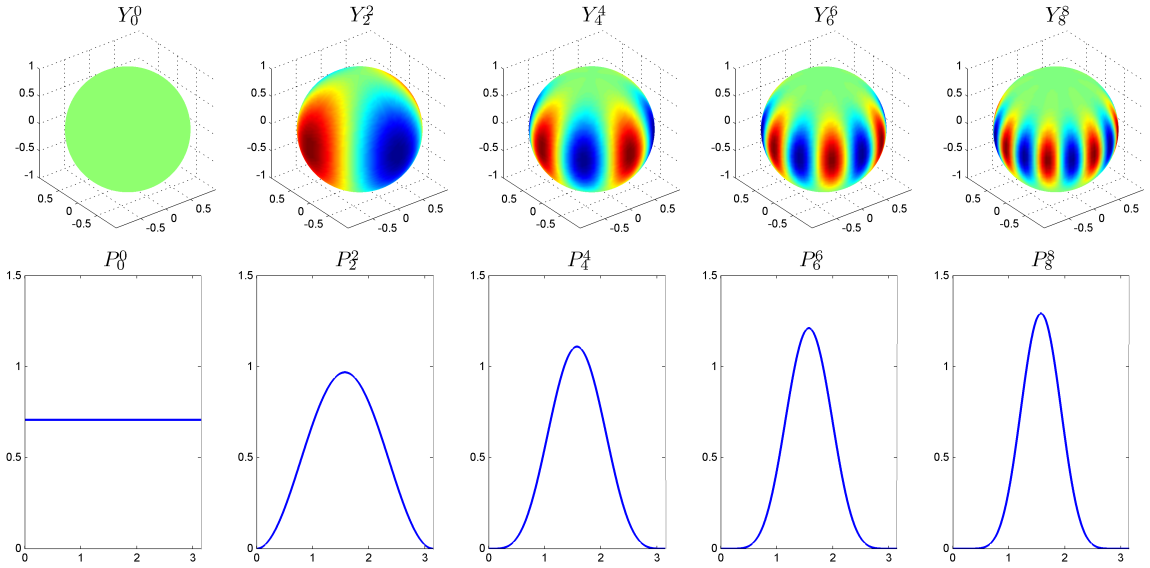

Figure 13: In the definition of the spherical harmonics $Y_l^m(\cdot)$, when $m = l$ there is no latitudinal dependence. As a result, only the harmonic frequencies in the longitudinal coordinate are computed. These are referred to as the sectoral harmonics shown in the top figures for several different harmonic degrees l. The bottom figures shows the contribution of the Legendre polynomials.

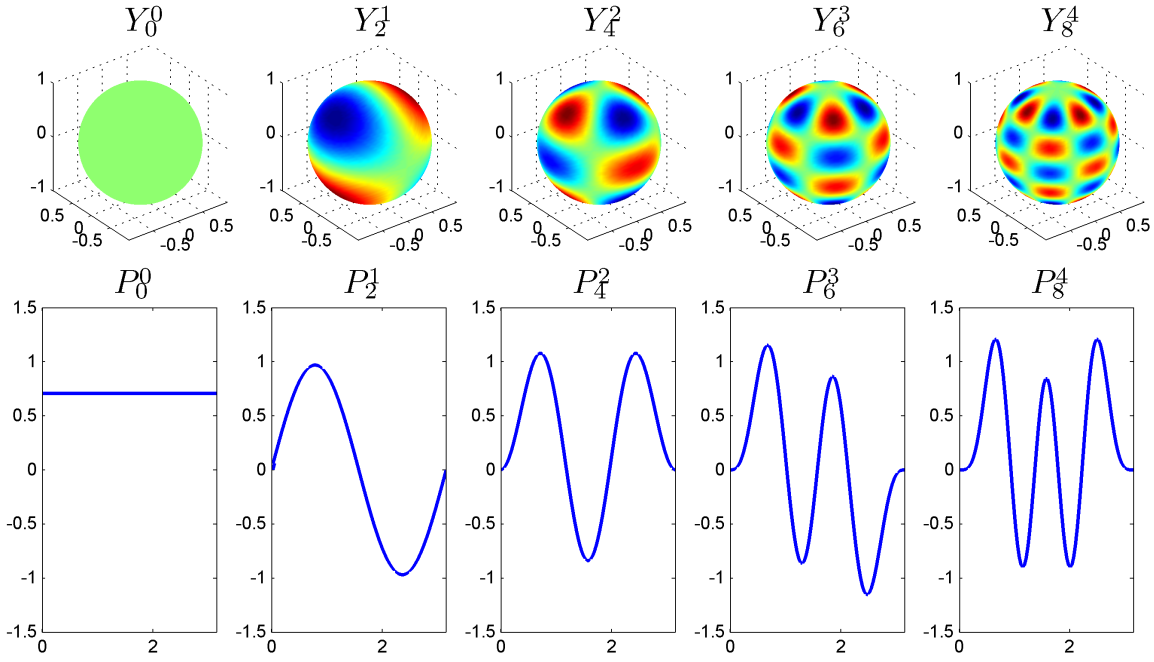

Figure 14: In the definition of the spherical harmonics $Y_l^m(\cdot)$, when $m \neq (0, l)$ harmonic frequencies in both the longitudinal and the co-latitudinal coordinate are computed. These are referred to as the tesseral harmonics shown in the top figures for several different harmonic degrees l. The bottom figures shows the contribution of the Legendre polynomials.

5.5 Spherical Harmonic Transform

Using the above development allows one to construct a transform for functions defined on the sphere (S^2) in much the same way the Fourier transform is defined for functions on the circle (S^1). Because the spherical harmonics provide an orthonormal basis for S^2, any square integrable function $f(\alpha, \beta) \in L^2(S^2)$ may be projected onto this basis as

$$f(\alpha, \beta) = \sum_{l=0}^{\infty} \sum_{|m| \leq l} f_l^m Y_l^m(\alpha, \beta) \tag{68}$$

with the expansion coefficients computed as

$$f_l^m = \int_0^{2\pi} \int_0^{\pi} f(\alpha, \beta) Y_l^m(\alpha, \beta)^* \sin(\beta) d\beta d\alpha, \tag{69}$$

where once again, * denotes the complex conjugate, and L^2 is the Hilbert space of square integrable functions.

Unfortunately, computing the harmonic coefficients by evaluating the integrals in Eq. (69) is prohibitively expensive. In order to reduce this computational expense, the integrals in (69) need to be approximated by finite sums and the development of a discrete spherical harmonic transform is used. The development of a discrete spherical harmonics transform is outlined in the following subsection.

5.5.1 Discrete Spherical Harmonic Transform

The discretization of integrals to finite sums has been addressed in various ways dating back to the 1800's [71]. The first step in this conversion is to determine the proper discretization of the sphere. The next step is to determine (if possible) a numerical quadrature in an attempt to make the discretization exact. Both of these steps were outlined in Chapter 4. As discussed in that chapter, for this particular application, the HEALPix discretization is superior and will be used to define a discrete spherical harmonic transform.

As mentioned in Chapter 2, the sphere may be parameterized by the total number of samples $n = ab$, where a is the number of samples defined on the sphere's surface (S^2), and b is the number of planar rotations captured at each sample. Using this parameterization, a function $f(\cdot)$ can sampled by

$$f(\cdot) = f(\boldsymbol{\xi}_p, \gamma_r), \tag{70}$$

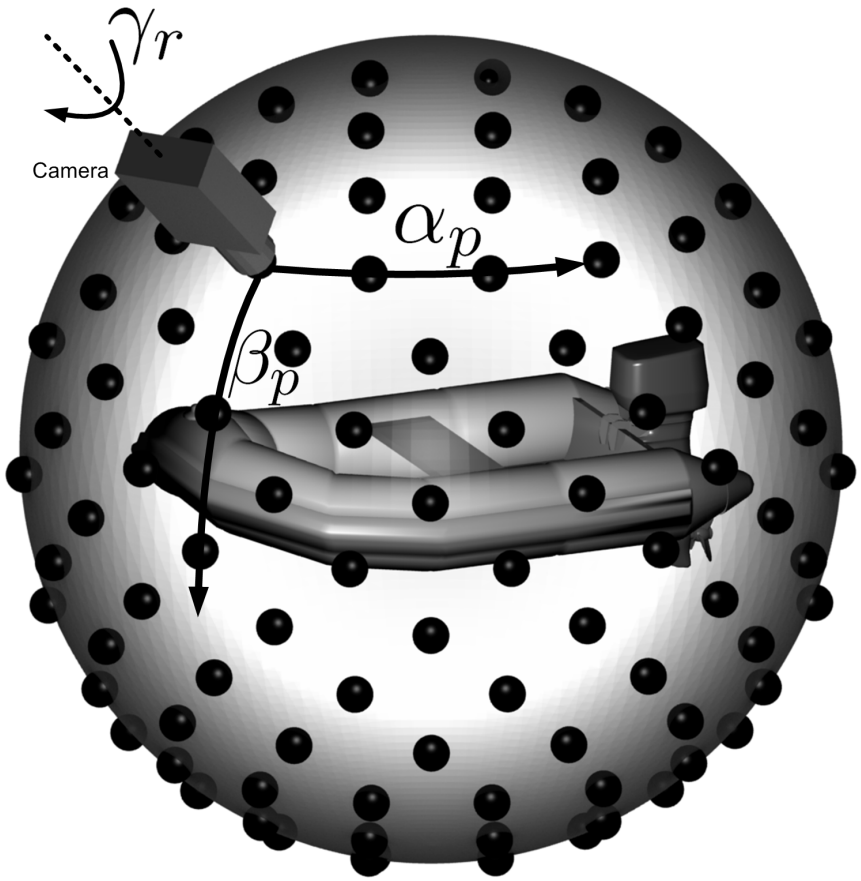

Figure 15: The proposed sampling method used in this dissertation with the parameterization of the sphere determined by (α, β, γ). Using this parameterization, extensions from S^1 to S^2 and finally, $SO(3)$ can be easily made.

where $\boldsymbol{\xi}_p$, $p \in \{0, \ldots, a-1\}$, is the unit vector pointing at the angle of co-latitude $\beta_p \in (0, \pi)$ measured down from the upper pole, and the angle of longitude $\alpha_p \in [0, 2\pi)$, which is the parameterization of the sphere in spherical coordinates. In $\boldsymbol{f}(\boldsymbol{\xi}_p, \gamma_r)$, the value $\gamma_r \in [0, 2\pi)$ is the r^{th} planar rotation at sample p where $r \in \{0, \ldots, b-1\}$. This parameterization is chosen so that the extensions from S^1 to S^2 and finally $SO(3)$ can be easily made.

Using the above parameterization, consider capturing images on the surface of the sphere where an object is placed at the sphere's center as shown in Fig. 15. Letting $r = \gamma_r = 0$, $\beta_p \in (0, \pi)$, and $\alpha_p \in [0, 2\pi)$ results in sampling the entire surface of the sphere rather than lines of constant latitude as in Chapter 3. Sampling the object in this manner results in a two-dimensionally correlated image data matrix that is correlated on S^2. A real valued

band-limited function $\boldsymbol{f}(\boldsymbol{\xi}_p, 0)$ whose domain is $L^2(S^2)$ may be represented by its discrete spherical harmonic expansion as

$$f(\boldsymbol{\xi}_p, 0) = \sum_{l=0}^{l_{\max}} \sum_{|m| \leq l} f_l^m Y_l^m(\boldsymbol{\xi}_p) \tag{71}$$

where $f(\boldsymbol{\xi}_p, 0) \in [0, 1]$ is a single pixel of the image data vector $\boldsymbol{f}(\boldsymbol{\xi}_p, 0)$. In the above equation, it is assumed that the signal power for $l > l_{\max}$ is insignificant, and $l_{\max}$ is chosen to prevent aliasing. The expansion coefficients are calculated using

$$f_l^m = \frac{4\pi}{n} \sum_{p=0}^{n-1} f(\boldsymbol{\xi}_p, 0) Y_l^m(\boldsymbol{\xi}_p)^*, \tag{72}$$

where $Y_l^m(\boldsymbol{\xi}_p)^*$ is the conjugate of the spherical harmonic of degree l and order m.

Because the functions in this work are real valued images, it is more convenient to use the real valued spherical harmonics. The real valued spherical harmonics are defined as

$$Y_l^m(\boldsymbol{\xi}_p) = \begin{cases} \sqrt{2}\kappa_l^m \cos(m\alpha_p) P_l^m(x) & \text{if } m > 0 \\ \sqrt{2}\kappa_l^m \sin(|m|\alpha_p) P_l^{|m|}(x) & \text{if } m < 0 \\ \kappa_l^0 P_l^0(\cos(\beta_p)) & \text{if } m = 0 \end{cases} \tag{73}$$

where $P_l^0(\cos(\beta_p)) = P_l(\cos(\beta_p))$ is the Legendre polynomial of degree l, and κ_l^m is the normalization constant

$$\kappa_l^m = \sqrt{\frac{2l+1}{4\pi} \frac{(l-m)!}{(l+m)!}}. \tag{74}$$

Using the real valued spherical harmonics, the expansion coefficients f_l^m can be computed using (72) by replacing $Y_l^m(\boldsymbol{\xi}_p)^*$ with its real counterpart.

5.6 Eigenspace Decomposition Algorithm on S^2

5.6.1 Motivation

In this section, a fast eigenspace decomposition algorithm based on the analysis of Section 5.5 is developed. In [26], Chang *et al.* showed that for image data sets correlated in one dimension, the right singular vectors are approximately spanned by the first few Fourier harmonics. While this is not true for image data sets that are spherically correlated in higher dimensions, most of the energy of these image data sets is concentrated around the lower frequency spherical harmonics. An example of this is shown in Fig. 16 where

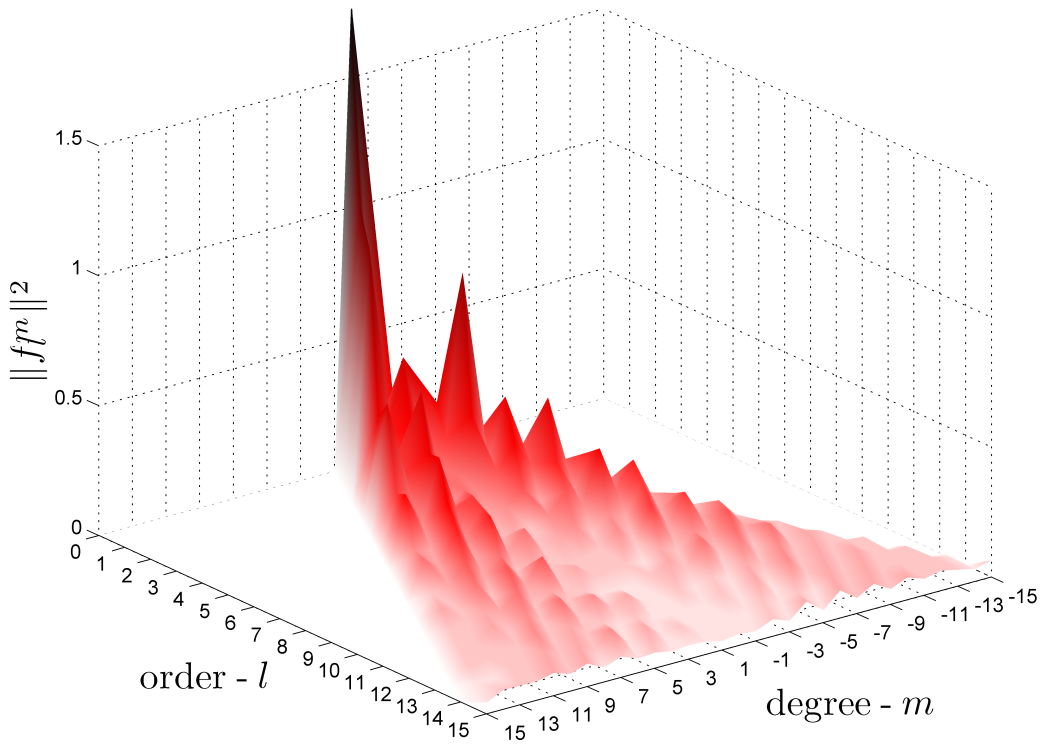

Figure 16: The spherical harmonic power spectra $||\boldsymbol{f}_l^m||^2$ of object (15) from Fig. 17. As can be seen from the figure, most of the energy is concentrated around the low frequency harmonics.

the spherical harmonic power spectra of object 15 from Fig. 17 is presented. As can be observed in the figure, in general, as l increases, the magnitude of the power spectra decreases. As a result, the left singular vectors $\tilde{U}_k$ of the SVD of a relatively small set of the spherically transformed harmonic images serve as excellent estimates to those of X, but can be computed at a significant computational savings [28, 31].

5.6.2 Algorithm

The objective is to estimate the desired subspace dimension k, as well as the principal eigenimages $\tilde{U}_k$ of X, so that $\rho(X, \tilde{U}_k) \geq \mu$, where μ is a user specified energy recovery ratio. The first step in computing the desired subspace dimension k, as well as the principal eigenimages, is to construct the image data matrix X. As mentioned previously, the approach taken here is to consider the object placed at the center of an imaginary unit sphere, and

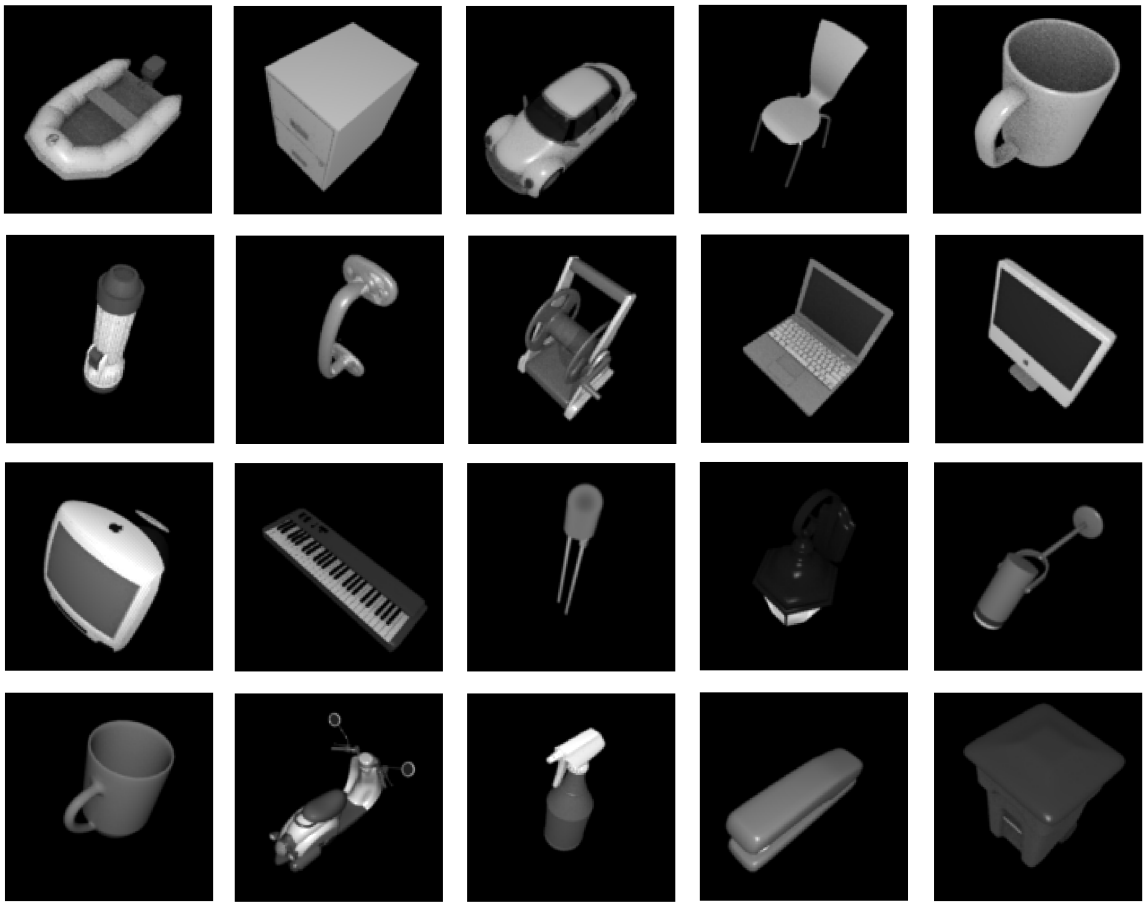

Figure 17: Example ray-traced images of CAD models courtesy of Kator Legaz [1]. Each object is sampled using the HEALPix sampling pattern at a resolution of 128×128. Each of the images are then scale and intensity normalized. The objects are ordered from left to right, then top to bottom.

then sample S^2 by capturing images of the object at every point of the HEALPix sampling pattern. The objects used in the study for the construction of X are CAD models courtesy of [1]. Each of the CAD models are ray-traced to generate one sample image, examples of which are shown in Fig. 17. The images are then both scale and intensity normalized to generate the image data matrix X.

Once the image data matrix X has been constructed, the matrix F is computed, whose i^{th} row is the SHT of the i^{th} row of X, denoted from this point forward as $\text{SHT}(X)$. This can be computed quickly using the methods described in [51], however for small bandwidths the computational savings of this method are slim. Alternatively, the $\text{SHT}(X)$ can be cast

as the matrix multiplication

$$F = XZ, \tag{75}$$

where $X \in \mathbb{R}^{m\times(12N_{\text{side}}^2)}$ is the image data matrix with the images ordered in terms of $(\boldsymbol{\xi}_p, 0)$ as

$$X = \left[\boldsymbol{f}(\boldsymbol{\xi}_0, 0), \boldsymbol{f}(\boldsymbol{\xi}_1, 0), \cdots, \boldsymbol{f}(\boldsymbol{\xi}_{12N_{\text{side}}^2-1}, 0)\right], \tag{76}$$

and $Z \in \mathbb{R}^{(12N_{\text{side}}^2)\times(9N_{\text{side}}^2)}$ is the matrix of spherical harmonics ordered in terms of $\boldsymbol{\xi}_p$, l, and m as

$$Z = \begin{bmatrix} Y_0^0(\boldsymbol{\xi}_0) & Y_1^{-1}(\boldsymbol{\xi}_0) & Y_1^0(\boldsymbol{\xi}_0) & Y_1^1(\boldsymbol{\xi}_0) & Y_2^{-2}(\boldsymbol{\xi}_0) & \cdots & Y_{3N_{\text{side}}-1}^{3N_{\text{side}}-1}(\boldsymbol{\xi}_0) \\ Y_0^0(\boldsymbol{\xi}_1) & & & \cdots & & & Y_{3N_{\text{side}}-1}^{3N_{\text{side}}-1}(\boldsymbol{\xi}_1) \\ \vdots & & & \vdots & & & \vdots \\ Y_0^0(\boldsymbol{\xi}_{12N_{\text{side}}^2-1}) & & & \cdots & & & Y_{3N_{\text{side}}-1}^{3N_{\text{side}}-1}(\boldsymbol{\xi}_{12N_{\text{side}}^2-1}) \end{bmatrix}. \tag{77}$$

Note that in computing the $\text{SHT}(X)$ the matrix Z may be pre-computed for several different bandwidths and stored for later use.

An algorithm is now presented for estimating the first k principal eigenimages $\tilde{U}_k$ of X such that $\rho(X, \tilde{U}_k) \geq \mu$, where μ is the user specified energy recovery ratio.

SHT-Based Eigenspace Decomposition Algorithm

1. Form the matrix F by computing the $\text{SHT}(X)$.
2. Form the matrix H whose columns are the ordered columns of F in descending order according to their norm.
3. Set $q = \lfloor (3N_{\text{side}})^2[1 - (1/2)^{N+1}]\rfloor$, with N=0 initially.
4. Construct the matrix H_q, i.e., the matrix consisting of the first q columns of H.
5. Compute $\text{SVD}(H_q) = \tilde{U}_q\tilde{S}_q\tilde{V}_q^T$. (The key observation here is that H_q contains q columns, which is considerably less than the n columns of X.)

6. If $\rho(X, \tilde{U}_q) < \mu$. Let $N = N + 1$ and repeat Steps 3 through 6. Because the SVD of H_q is already available, the eigenspace can simply be updated by modifying the algorithm outlined in [39].

7. Return $\tilde{U}_k$ such that $\rho(X, \tilde{U}_k) \geq \mu$. Note that $k \leq q$.

The above algorithm takes advantage of the fact that most SVD algorithms require mn^2 flops to compute the full SVD of X; this is computationally prohibitive when n is large. Because the SHT(X) is lossy, step one of the algorithm condenses the energy in X from $12N_{\text{side}}^2$ images to $9N_{\text{side}}^2$ harmonic images ordered in terms of the magnitude of their power spectra to form the matrix H. Furthermore, because most of the energy of X is concentrated around the lower frequency spherical harmonics, the matrix H_q is constructed and the SVD is performed on the first $q = \lfloor (3N_{\text{side}})^2[1 - (1/2)^{N+1}] \rfloor$ harmonic images initially, significantly reducing the computational cost. If more harmonics are required, then half of the remaining harmonic images are concatenated to H_q and the eigenspace is updated.

It is difficult to evaluate the computational complexity of the above algorithm due to the binary split used to determine k. However, if only one iteration of steps 3-6 is performed, i.e., $N = 0$, then the complexity is on the order of mq^2 flops where $q = \lfloor (3N_{\text{side}})^2[1-(1/2)^{N+1}] \rfloor$. This is the cost of computing the SVD(H_q) in step 5 of the algorithm. It should be noted that for all of the objects tested over 90% of the energy in X was recovered with a single itteration. Futhermore, extensive simulation has shown that for pose estimation, a subspace that can recover between 60% - 70% of the energy in X is typically more than sufficient [27, 28].

5.7 Experimental Results

5.7.1 Test Data

The proposed algorithm detailed in Section 5.6 was tested on the objects shown in Fig. 17. The parameter $N_{\text{side}} = 8$ was used, resulting in $12N_{\text{side}}^2 = 768$ images per object at an angular resolution $\approx 7^\circ$. The images were then both scale and intensity normalized to create the image data matrix X. Finally, the matrix F was computed condensing the

image data set from 768 images to 576 harmonic images. The true SVD(X) was computed using MATLAB for a comparison.

5.7.2 Quality of Estimation

Fig. 18 shows a plot of the difference between the energy recovery ratio using the true left singular vectors ρ_t, and the energy recovery ratio using the proposed left singular vectors computed using the proposed algorithm ρ_p, as a function of the subspace dimension k averaged across all objects in Fig. 17. As can be seen from the figure, there is less than a 0.05% difference in the energy recovered using the proposed algortihm as compared to the true eigenimages, i.e., the estimated left singular vectors $\tilde{U}_k$ as computed by the proposed algorithm are very good estimates of the true left singular vectors as computed by the SVD(X) in terms of being able to recover the energy in X. Note, this data is for the first split, i.e., $N = 0$, in step 3 of the algorithm. and there is less than a 0.05% difference in the energy recovered using the proposed algortihm.

5.7.3 Computational Savings

Table 3 shows the computation time required to calculate the subspace dimension k, and estimate the left singular vectors $\tilde{U}_k$ required to meet the user specified energy recovery ratio of $\mu = 0.9$. Also depicted in the table is the result as computed by the true SVD(X). As is apparent from the table, in most cases, the left singular vectors $\tilde{U}_k$ are very good estimates

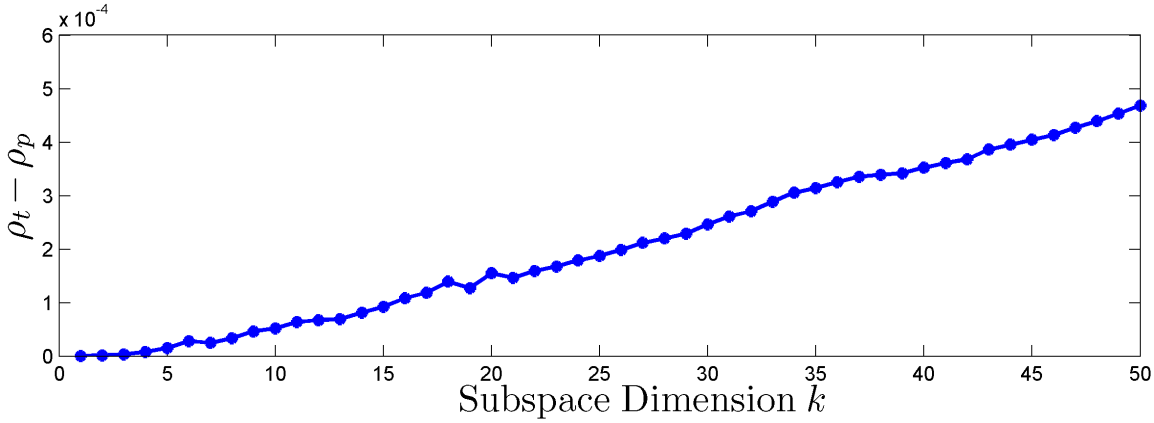

Figure 18: Difference between the true and proposed energy recovery ratio vs. subspace dimension k averaged across all 20 objects in Fig. 17.

Table 3: The required subspace dimension k, and the time required to estimate the first k left singular vectors for each object in Fig. 17 to exceed the user specified energy recovery $\mu = 0.9$. The results are compared against the true SVD using MATLAB.

Object no.	Dimension k		Time [sec.]	
	True	Proposed	True	Proposed
1	18	18	24.36	7.03
2	3	3	23.89	6.70
3	21	21	24.71	7.13
4	67	71	25.97	8.34
5	5	5	24.00	6.92
6	3	3	26.38	6.67
7	26	26	24.27	7.20
8	60	61	25.81	8.08
9	22	22	24.77	7.16
10	35	36	29.87	7.48
11	26	26	24.62	7.25
12	47	48	25.12	7.76
13	8	8	23.34	6.77
14	19	20	24.79	7.28
15	47	49	25.31	7.81
16	3	3	23.38	6.69
17	54	56	29.93	7.95
18	11	11	23.82	6.87
19	12	12	31.95	6.94
20	2	2	27.44	6.71

of U_k at a significant computational savings using the proposed algorithm. Again, only the first split, i.e., $N = 0$, in step 3 of the algorithm is required to recover over 90% of the energy as depicted in the table.

5.7.4 Maximum Energy Recovery

Based on the data provided in Table 3 and Fig. 18, it is obvious that the proposed algorithm is capable of estimating the required subspace dimension and left singular vectors of X at a significant computational savings. Because the SHT is lossy however, it is important to quantify the maximum amount of energy recovery possible using this approach. To do this, all $9N_{\text{side}}^2$ harmonic images were used to estimate the left singular vectors of X. Using these estimates, the energy recovery ratio $\rho(X, \tilde{U}_{9N_{\text{side}}^2})$ was computed for each object in Fig. 17. The top plot in Fig. 19 shows the maximum amount of energy recovered per object. The bottom plot shows the maximum difference in energy recovery per object if only the first

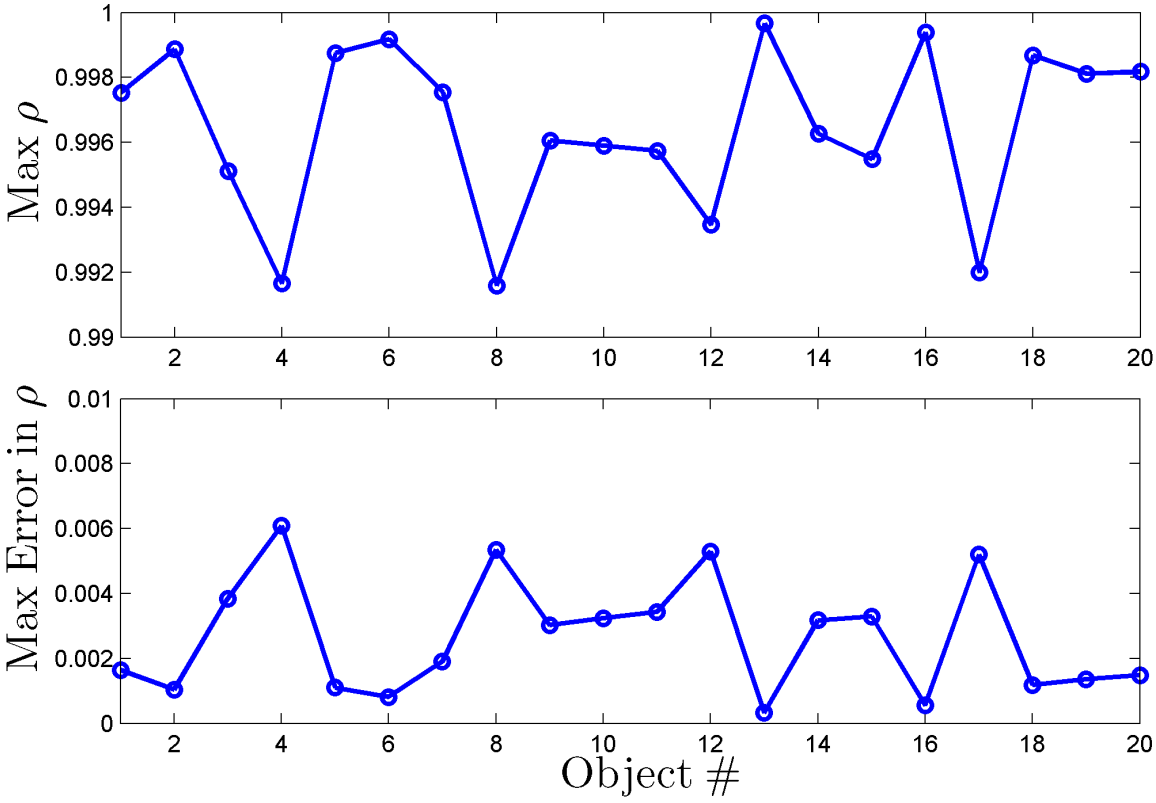

Figure 19: The maximum achievable energy recovery ratio ρ for each object in Fig. 17 [top], and the maximum difference in energy recovery between the first $9N_{\text{side}}^2$ true left singular vectors as computed by SVD(X) and the estimated left singular vectors $U_{9N_{\text{side}}^2}$ as computed by the proposed algorithm [bottom].

$9N_{\text{side}}^2$ true left singular vectors are used to recover the same energy, i.e., the bottom plot shows $\max(\rho(X, U_k) - \rho(X, \tilde{U}_k))$ for all $k \leq 9N_{\text{side}}^2$ for each object. As can be seen from the figure over 99% of the energy in X is recoverable for all objects using the proposed algorithm. Furthermore, there is less than a 1% difference between the energy recovered by the first $9N_{\text{side}}^2$ true left singular vectors and the left singular vectors computed by the proposed algorithm for all objects in Fig. 17. Finally, using

$$\frac{1}{4.5N_{\text{side}}^2} \sum_{i=2}^{4.5N_{\text{side}}^2} [\rho(X, U_i) - \rho(X, \tilde{U}_i)] \tag{78}$$

the average error in energy recovery is computed for each object in Fig. 17. The results are depicted in the top plot of Fig. 20, which shows that the average error one could expect to see for the objects of Fig. 17 is less than 1%. Note that $4.5N_{\text{side}}^2$ is chosen in the summation of (78) because this is the number of harmonic images resulting in the first split (step 3)

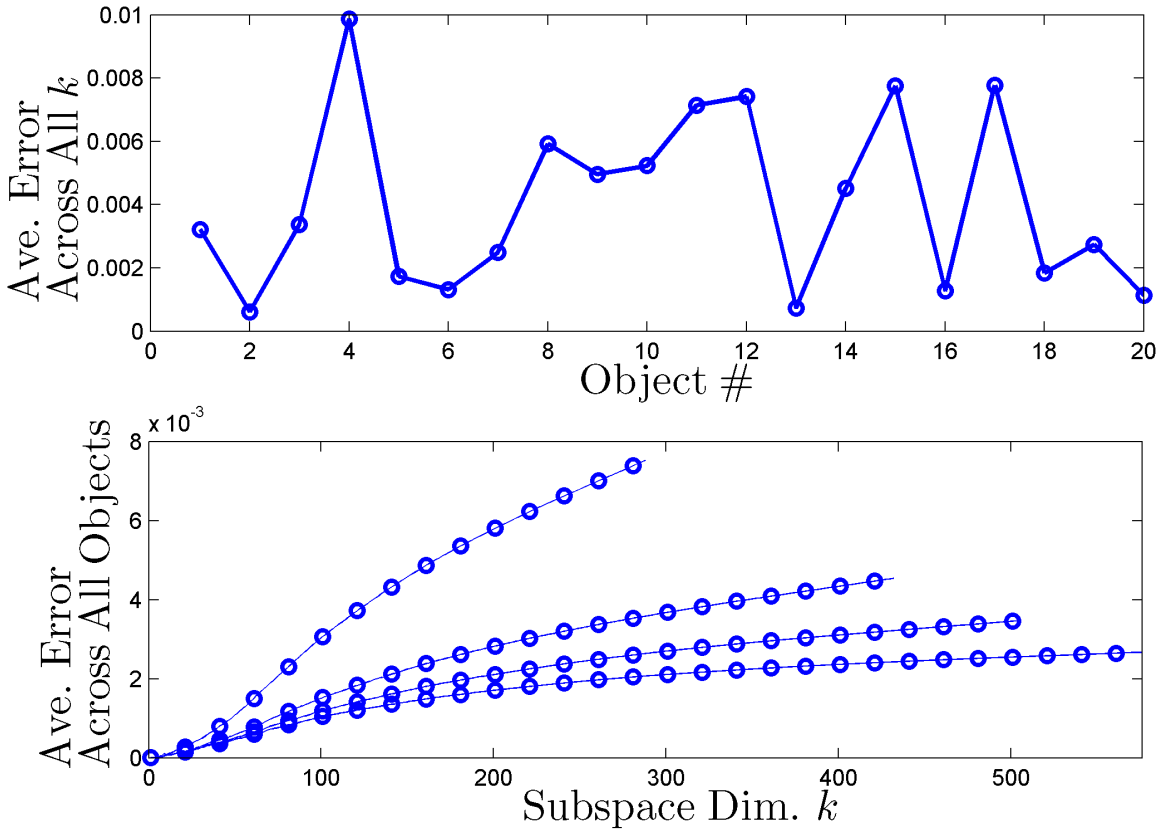

Figure 20: The average error in energy recovery using the true left singular vectors as computed by SVD(X), and the approximated left singular vectors calculated using the proposed algorithm for each object in Fig. 17 across the subspace dimension k [top]. The bottom plot shows the error in energy recovery averaged across all objects in Fig. 17 using the approximated left singular vectors calculated by the first four splits (step 3) of the proposed algorithm.

of the proposed algorithm. The bottom plot in Fig. 20 shows the error as a function of subspace dimension averaged across all objects in Fig. 17. Note that the first four splits are shown and that there is less than 0.8% error across all objects regardless of the number of splits used. Furthermore, adding more harmonic images to H_q usually results in less than 1/1000 less error. This further supports the claim that for most objects, most of the energy of X is concentrated around the low frequency spherical harmonics.

5.8 Hemispherical Harmonics

As a brief aside, it is shown that with a slight modification to the above development, pose detection from an aerial perspective can be achieved [27]. This detour is motivated by the

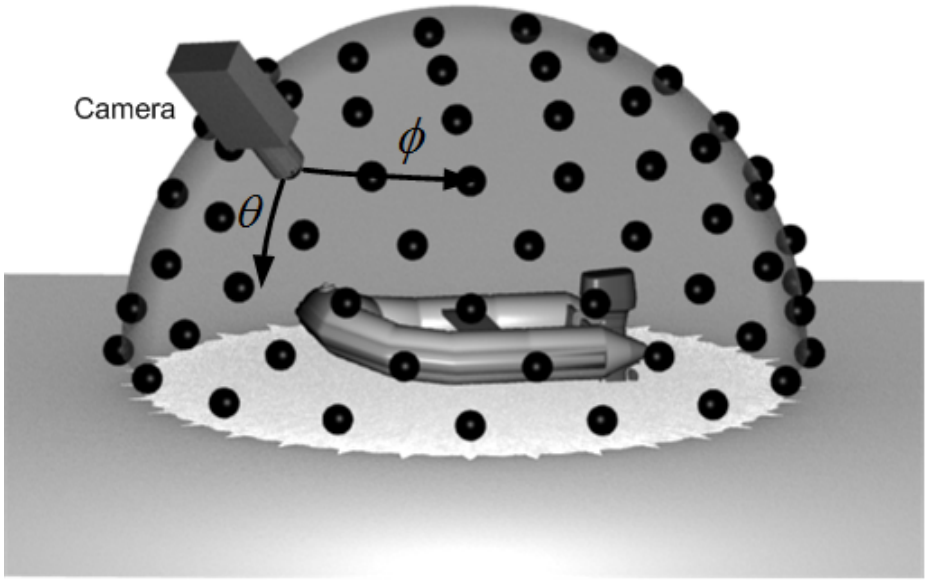

Figure 21: Image acquisition for the training image data matrix X using hemispherical sampling for aerial pose estimation. A sample image is taken at each black dot on the hemisphere.

fact that numerous pose detection applications are only interested in estimating the pose of three-dimensional objects from an aerial view. Some examples include aerial surveillance and reconnaissance, remote sensing, and part inspection.

Consider sampling the sphere as discussed in Section 5.6, however restricting the angle of co-latitude to $\beta_p \in (0, \pi/2)$ (an example of this is depicted in Fig. 21). Applying this constraint then restricts the sampling to the upper hemisphere of S^2. It has been shown that applying a linear shift of $x = \cos(\beta_p)$ to $x = 2\cos(\beta_p - 1)$ in the associated Legendre polynomials, and redefining the normalization constant κ_l^m, the spherical harmonics form an orthonormal basis on the upper hemisphere of S^2 [27, 72]. Therefore, a real valued band-limited function $f(\boldsymbol{\xi}_p, 0)$ whose domain is the upper hemisphere of $L^2(S^2)$ can be represented by its hemispherical harmonic expansion as

$$f(\boldsymbol{\xi}_p, 0) = \sum_{l=0}^{l_{\max}} \sum_{|m| \leq l} f_l^m H_l^m(\boldsymbol{\xi}_p). \tag{79}$$

The expansion coefficients are calculated by

$$f_l^m = \frac{2\pi}{n} \sum_{p=0}^{n-1} f(\boldsymbol{\xi}_p, 0) H_l^m(\boldsymbol{\xi}_p), \tag{80}$$

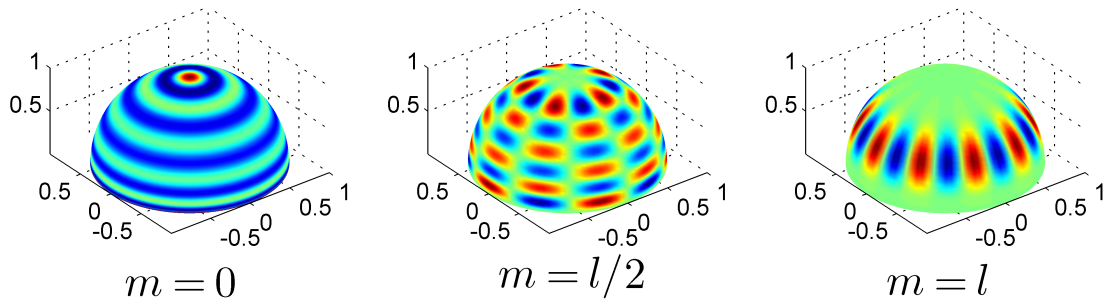

Figure 22: The real hemispherical harmonics $H_l^m(\cdot)$ for $l = 8$. The left plot is for $m = 0$, the center plot is for $m = l/2$, and the right plot is for $m = l$.

where $H_l^m(\boldsymbol{\xi}_p)$ is the real-valued hemispherical harmonic defined by

$$H_l^m(\boldsymbol{\xi}_p) = \begin{cases} \sqrt{2}\bar{\kappa}_l^m \cos(m\alpha_p)\bar{P}_l^m(x) & \text{if } m > 0 \\ \sqrt{2}\bar{\kappa}_l^m \sin(|m|\alpha_p)\bar{P}_l^{|m|}(x) & \text{if } m < 0 \\ \bar{\kappa}_l^0 \bar{P}_l^0(x) & \text{if } m = 0 \end{cases} \tag{81}$$

with $\bar{P}_l^m(x)$ being a shifted associated Legendre polynomial of degree l and order m, $x = 2\cos(\beta_p) - 1$, and

$$\bar{\kappa}_l^m = \sqrt{\frac{2l+1}{2\pi}\frac{(l-|m|)!}{(l+|m|)!}} \tag{82}$$

is the normalization constant for the hemispherical transform. Examples of the real hemispherical harmonics projected onto the hemisphere for $l = 8$ and three different values of m are shown in Fig. 22. The eigenspace decomposition algorithm developed in 5.6 is easily extended to computing the eigendecompositon of hemispherically correlated images by replacing step 1 with "Form the matrix F by computing the HSHT(X)", and step 3 with "Set $q = \lfloor (1.5N_{\text{side}})^2[1 - (1/2)^{N+1}] \rfloor$, with N=0 initially" [27].

CHAPTER VI

WIGNER-D MATRICES AND THE ROTATION GROUP

6.1 Chapter Overview

In this chapter the spherical harmonic transform developed in Chapter 5 is extended to the rotation group $SO(3)$ using some of the techniques developed in [73]. This extension is necessary for the development of a fully general 3-D pose estimation algorithm. Section 6.2 begins by showing how the spherical harmonic transform developed in Chapter 5 can be extended to the full rotation group $SO(3)$ using Wigner-D matrices. Using this extension, an $SO(3)$ harmonic transform is developed in Section 6.3. This transform provides the necessary means to extend the algorithm developed in Chapter 5 from computing the eigenspace decomposition of an image data set correlated on S^2 to an image data set correlated on $SO(3)$. The construction of this algorithm is presented in Section 6.4. Finally, this algorithm is applied to fully general 3-D data sets in Section 6.5, and an analysis is presented illustrating its effectiveness in estimating the eigenspace of arbitrary 3-D objects.

6.2 Introduction

For the construction of the $SO(3)$ FFT the rotation of a function defined on S^2 is computed using elements of the rotation group. The rotation group is the set of real 3×3 orthogonal matrices of determinant $+1$, which define proper rotations about the origin of $\mathbb{R}^3$. In spectral theory, it is often the convention to define these matrices using standard $z - y - z$ Euler rotation matrices where the z-axis is the upper pole [57, 58, 74]. Therefore, any rotation $g(\alpha, \beta, \gamma) \in SO(3)$ can be written as

$$g(\alpha, \beta, \gamma) = R_z(\alpha) R_y(\beta) R_z(\gamma) \tag{83}$$

where $R_z(\alpha)$ and $R_y(\beta)$ represent a rotation about the z-axis by α radians, and a rotation about the y-axis by β radians respectively. Given any $g \in SO(3)$, the linear operator $\Lambda(\alpha, \beta, \gamma) : f(\alpha, \beta) \rightarrow f(\alpha', \beta')$ is defined where (α, β) and (α', β') are the coordinates of the position vector in the original and rotated coordinate frames respectively. The effect this has on the function in the spectral domain (i.e., the effect on the harmonic coefficient f_l^m) can be deduced from the fact that rotated versions of the spherical harmonics are simply linear combinations of harmonics of the same degree. That is

$$\Lambda(\alpha, \beta, \gamma) Y_l^m(\alpha, \beta) = Y_l^m(\alpha', \beta') = \sum_{|m| \leq l} Y_l^m(\alpha, \beta) D_{mm'}^l(\alpha, \beta, \gamma) \tag{84}$$

where $D_{mm'}^l(\alpha, \beta, \gamma)$ is the $(2l+1) \times (2l+1)$ Wigner-D matrix [57].

6.3 Harmonic Analysis on $SO(3)$

Using the Peter-Weyl theorem, it can be shown that the D matrices satisfy the condition

$$\int_0^{2\pi} \int_0^{\pi} \int_0^{2\pi} D_{m_1 m_1'}^{l_1 *}(\alpha, \beta, \gamma) D_{m_2 m_2'}^{l_2}(\alpha, \beta, \gamma) d\Omega = \frac{8\pi^2}{2l_1 + 1} \delta_{l_1 l_2} \delta_{m_1 m_2} \delta_{m_1' m_2'}, \tag{85}$$

where $d\Omega = d\alpha \sin(\beta) d\beta \, d\gamma$, i.e., they form an orthogonal basis over $SO(3)$ [73]. Therefore, any function $f(\alpha, \beta, \gamma) \in L^2(SO(3))$ may be projected onto this basis as

$$f(\alpha, \beta, \gamma) = \sum_{l=0}^{\infty} \sum_{|m| \leq l} \sum_{|m'| \leq l} f_{mm'}^l D_{mm'}^l(\alpha, \beta, \gamma) \tag{86}$$

with the expansion coefficients computed as

$$f_{mm'}^l = \frac{2l+1}{8\pi^2} \int_0^{2\pi} \int_0^{\pi} \int_0^{2\pi} f(\alpha, \beta, \gamma) D_{mm'}^{l *}(\alpha, \beta, \gamma) d\Omega, \tag{87}$$

where once again, * denotes the complex conjugate and L^2 is the Hilbert space of square integrable functions.

Unfortunately, as in Chapter 5, the integrals in (87) must be approximated by finite sums and the development of a discrete $SO(3)$ harmonic transform is needed. The development of a discrete $SO(3)$ harmonic transform is outlined in the following subsection.

6.3.1 Discrete $SO(3)$ Harmonic Transform

Before developing a discrete harmonic transform on $SO(3)$, the reader is reminded that $SO(3)$ can be discretized by sampling the sphere using the HEALPix sampling pattern.

This results in capturing $n = ab$ samples, where a is the number of samples defined on the sphere's surface (S^2), and b is the number of planar rotations captured at each sample. Using this parameterization, a function $f(\cdot)$ can be sampled by

$$f(\cdot) = f(\boldsymbol{\xi}_p, \gamma_r), \tag{88}$$

where $\boldsymbol{\xi}_p$, $p \in \{0, \ldots, a-1\}$, is the unit vector pointing at the angle of co-latitude $\beta_p \in (0, \pi)$ measured down from the upper pole, and the angle of longitude $\alpha_p \in [0, 2\pi)$, which is the parameterization of the sphere in spherical coordinates. In $\boldsymbol{f}(\boldsymbol{\xi}_p, \gamma_r)$, the value $\gamma_r \in [0, 2\pi)$ is the r^{th} planar rotation at sample p where $r \in \{0, \ldots, b-1\}$.

Using the above parameterization, consider capturing images on the surface of the sphere where an object is placed at the sphere's center as shown in Fig. 15. Letting $\gamma_r \in [0, 2\pi)$, $\beta_p \in (0, \pi)$, and $\alpha_p \in [0, 2\pi)$ results in sampling $SO(3)$, rather than lines of constant latitude as in Chapter 3 or the surface of the sphere as in Chapter 5. Sampling the object in this manner results in a three-dimensionally correlated image data matrix that is correlated on $SO(3)$.

Using (84) and (85), it can be shown that a function $f \in L^2(SO(3))$ can be represented by its discrete harmonic expansion using Wigner-D matrices [67, 73]. That is,

$$f(\boldsymbol{\xi}_p, \gamma_r) = \sum_{l=0}^{l_{\max}} \sum_{|m| \leq l} \sum_{|m'| \leq l} f_{mm'}^l D_{mm'}^l(\boldsymbol{\xi}_p, \gamma_r) \tag{89}$$

where once again, $f(\boldsymbol{\xi}_p, \gamma_r) \in [0, 1]$ is a single pixel of the image data vector $\boldsymbol{f}(\boldsymbol{\xi}_p, \gamma_r)$. The expansion coefficients $f_{mm'}^l$ are then calculated using

$$f_{mm'}^l = \frac{4\pi}{a} \sum_{p=0}^{a-1} \sum_{r=0}^{b-1} f(\boldsymbol{\xi}_p, \gamma_r) D_{mm'}^{l*}(\boldsymbol{\xi}_p, \gamma_r) \tag{90}$$

where * denotes complex conjugation. In the above equations, with the unit vector $\boldsymbol{\xi}_p$ parameterized by (α_p, β_p),

$$D_{mm'}^l(\alpha_p, \beta_p, \gamma_r) = e^{-im\alpha_p} d_{mm'}^l(\beta_p) e^{-im'\gamma_r} \tag{91}$$

where $d_{mm'}^l(\beta_p)$ is known as Wigner's (small) d-matrix defined by

$$d_{mm'}^l(\beta_p) = \sqrt{\frac{(l+m')!(l-m')!}{(l+m)!(l-m)!}} \left(\sin\frac{\beta_p}{2}\right)^{m'-m} \left(\cos\frac{\beta_p}{2}\right)^{m+m'} P_{(m'-m, m+m')}^{(l-m')}(\cos\beta_p) \tag{92}$$

and $P^{(\cdot)}_{(\cdot,\cdot)}(x)$ is a Jacobi polynomial. For computational convenience, the d-matrices may be computed quickly using a three term recurrence relationship [57, 73, 75].

Because $f(\boldsymbol{\xi}_p, \gamma_r)$ is a real-valued function, it is more convenient to work with the real-valued Wigner-D matrices denoted here as $\Delta^l_{mm'}$. The construction of rotation matrices in the basis of real spherical harmonics is discussed in [75–77], and can be defined as

$$\begin{aligned}\Delta^l_{mm'} = & \operatorname{sign}(m')\Phi_m(\alpha_p)\Phi_{m'}(\gamma_r)\frac{d^l_{|m'||m|}(\beta_p)+(-1)^m d^l_{|m|(-|m'|)}(\beta_p)}{2} \\ & -\operatorname{sign}(m)\Phi_{-m}(\alpha_p)\Phi_{-m'}(\gamma_r)\frac{d^l_{|m'||m|}(\beta_p)-(-1)^m d^l_{|m|(-|m'|)}(\beta_p)}{2}\end{aligned} \tag{93}$$

where

$$\Phi_m(x) = \begin{cases} \sqrt{2}\cos(mx) & \text{if } m > 0 \\ 1 & \text{if } m = 0 \\ \sqrt{2}\sin(|m|x) & \text{if } m < 0 \end{cases}. \tag{94}$$

The $SO(3)$ FFT may then be computed using (90) by replacing $D^{l*}_{mm'}(\boldsymbol{\xi}_p, \gamma_r)$ with $\Delta^l_{mm'}(\boldsymbol{\xi}_p, \gamma_r)$.

To show the harmonic extension from S^1 to S^2 and finally $SO(3)$, the separation of variables technique discussed in [73] is used along with the definition of the Wigner-D matrices given in (91). Applying these to the summations of (90), the $SO(3)$ Fourier coefficients can be computed as

$$f^l_{mm'} = \frac{4\pi}{a}\sum_{p=0}^{a-1} d^l_{mm'}(\beta_p)\sum_{r=0}^{b-1} e^{im'\gamma_r}\sum_{p=0}^{a-1} e^{im\alpha_p} f(\boldsymbol{\xi}_p, \gamma_r). \tag{95}$$

Note that the last summation is only concerned with data along lines of constant latitude, and is equivalent to computing the harmonic coefficients of data correlated on S^1 using the DFT. The second summation is equivalent to the special case discussed in Chapter 3, and can be computed quickly using FFT techniques. Finally, the first summation computes the harmonic coefficients of the co-latitudinal coordinate thus completing the full $SO(3)$ FFT. Notice that if only the first and last summation are computed (or equivalently $m' = 0$), then the transform is equivalent to computing the discrete spherical harmonic transform for a function correlated on S^2. In fact, it can be shown that by setting $m' = 0$, the Wigner-D matrices are equivalent to the spherical harmonics [57].

6.4 Eigenspace Decomposition Algorithm on $SO(3)$

6.4.1 Motivation

In this section a fast eigenspace decomposition algorithm is developed based on the analysis of Section 6.3, and is motivated by Chang's algorithm, as well as the algorithm developed in Chapter 5. In [26], Chang *et al.* showed that for image data sets correlated in one dimension, the right singular vectors are approximately spanned by the first few Fourier harmonics. While this is not true for image data sets that are spherically correlated in higher dimensions, it was shown in Chapter 5 that for image data sets correlated on S^2, most of the energy is concentrated around the lower frequency spherical harmonics. For objects correlated on $SO(3)$, most of the energy information is concentrated around the low frequency $SO(3)$ harmonics. This can be observed in Fig. 23, which is the $SO(3)$ power spectra for object 15 from Fig. 17. As the figure depicts, in general, as l increases the magnitude of the power spectra decreases. As a result, the left singular vectors $\tilde{\hat{U}}_k$ of the SVD of a relatively small set of the $SO(3)$ spherically transformed harmonic images serve as excellent estimates to those of $\hat{X}$ at a significant computational savings [29, 31].

6.4.2 Algorithm

Similar to the algorithm developed in Chapter 5 for images correlated on S^2, the objective is to estimate the first k principal eigenimages $\tilde{\hat{U}}_k$ of $\hat{X}$ such that $\Delta\rho(\hat{X}, \tilde{\hat{U}}_k) \leq \epsilon$, where ϵ is the user specified change in energy. The first step in computing the principal eigenimages is to construct the image data matrix $\hat{X}$ by sampling $SO(3)$ appropriately using the HEALPix sampling pattern for the discretization of the sphere. The mean image is then subtracted from each sample image to form the image data matrix $\hat{X}$.

As discussed in Chapter 4, using the HEALPix sampling pattern is based on subdividing the sphere using the parameter N_{side}, resulting in $a = 12N_{\text{side}}^2$ sample points on the sphere [52]. At each of the $12N_{\text{side}}^2$ sample points, b planar rotated images of the object are captured by rotating the camera through an angle γ_r. The number of planar rotated images captured depends on the angular resolution of the $12N_{\text{side}}^2$ samples defined on S^2. The

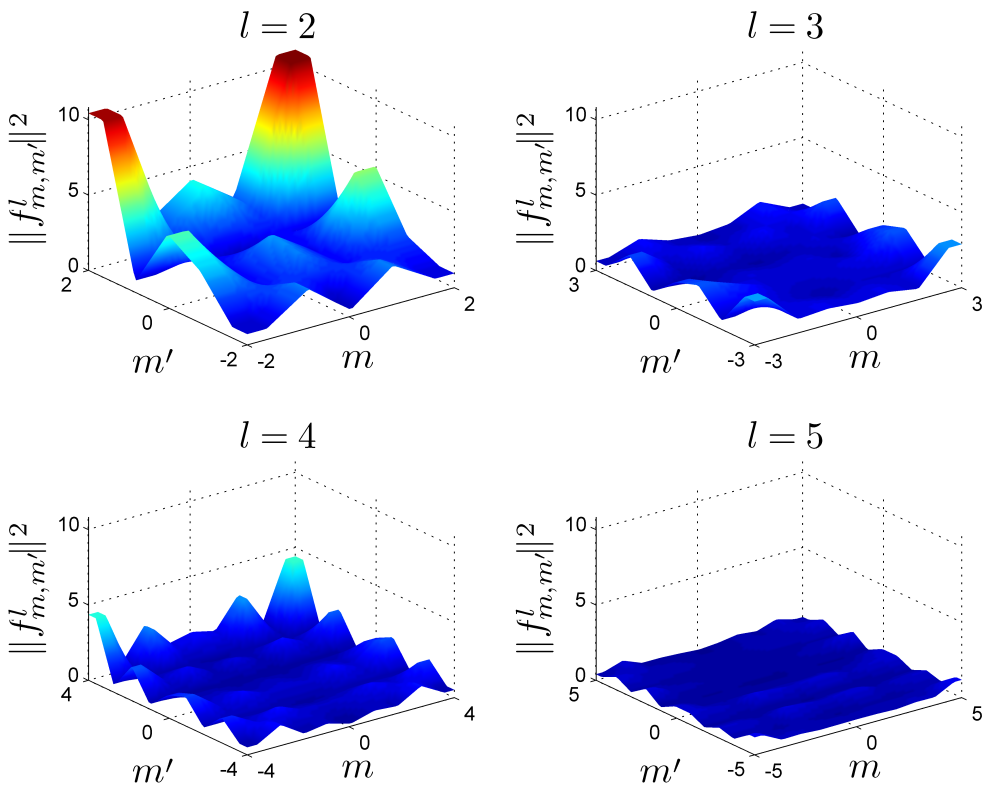

Figure 23: The power spectra $||\boldsymbol{f}^l_{mm'}||^2$ of object (15) from Fig. 17 for degrees $l = 2, 3, 4, 5$. As can be seen from the figure, most of the energy is concentrated around the lower frequency harmonics, i.e. those with lower values of l. ($l = 0$ is omitted as this represents the mean value and is significantly higher in magnitude.)

angular resolution is denoted θ_{pix} and calculated as $\theta_{\text{pix}} = \sqrt{\frac{3}{\pi}}\frac{60^o}{N_{\text{side}}}$. To maintain homogeneous sampling on $SO(3)$, $b = \lfloor 360/\theta_{\text{pix}} \rfloor$. Finally, to prevent aliasing, $l_{\max} = 3N_{\text{side}} - 1$ is used in the forward transform. Because the HEALPix sampling pattern is isolatitudinal, the computation of the Wigner-d matrices (which is the most computationally expensive portion of the $SO(3)$ FFT) is minimal.

Once the image data matrix $\hat{X}$ has been constructed, the matrix F (whose i^{th} row is the $SO(3)$ FFT of the i^{th} row of $\hat{X}$) is computed, denoted from this point forward as $SOFT(\hat{X})$. This can be computed quickly using the method described in [73], however for small $l_{\max}$ the computational savings of this method are slim. Alternatively, $SOFT(\hat{X})$ can be cast

as the matrix multiplication

$$F = \hat{X}Z, \tag{96}$$

where $\hat{X} \in \mathbb{R}^{m \times n}$ is the image data matrix with the image mean removed and the images ordered in terms of $(\boldsymbol{\xi}_p, \gamma_r)$ as

$$\begin{aligned} \hat{X} \;=\; & [\boldsymbol{f}(\boldsymbol{\xi}_0, \gamma_0), \boldsymbol{f}(\boldsymbol{\xi}_0, \gamma_1), \cdots, \boldsymbol{f}(\boldsymbol{\xi}_0, \gamma_{b-1}), \boldsymbol{f}(\boldsymbol{\xi}_1, \gamma_0), \boldsymbol{f}(\boldsymbol{\xi}_1, \gamma_1), \cdots, \\ & \boldsymbol{f}(\boldsymbol{\xi}_1, \gamma_{b-1}), \cdots, \boldsymbol{f}(\boldsymbol{\xi}_{a-1}, \gamma_0), \boldsymbol{f}(\boldsymbol{\xi}_{a-1}, \gamma_1), \cdots, \boldsymbol{f}(\boldsymbol{\xi}_{a-1}, \gamma_{b-1})]. \end{aligned} \tag{97}$$

The matrices $\Delta^l_{mm'}(\cdot) \in \mathbb{R}^{(2l+1)\times(2l+1)}$ can be row scanned such that each row is concatenated to form the row vector $\boldsymbol{\delta}^l(\cdot) = \mathrm{vec}(\Delta^l_{mm'}(\cdot)) \in \mathbb{R}^{1\times(2l+1)^2}$ for any given l. Using this notation, the matrix Z can be constructed as

$$Z = \begin{bmatrix} \boldsymbol{\delta}^0(\boldsymbol{\xi}_0, \gamma_0) & \boldsymbol{\delta}^1(\boldsymbol{\xi}_0, \gamma_0) & \cdots & \boldsymbol{\delta}^{3N_{\mathrm{side}}-1}(\boldsymbol{\xi}_0, \gamma_0) \\ \boldsymbol{\delta}^0(\boldsymbol{\xi}_0, \gamma_1) & \boldsymbol{\delta}^1(\boldsymbol{\xi}_0, \gamma_1) & \cdots & \boldsymbol{\delta}^{3N_{\mathrm{side}}-1}(\boldsymbol{\xi}_0, \gamma_1) \\ \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{\delta}^0(\boldsymbol{\xi}_0, \gamma_{b-1}) & \boldsymbol{\delta}^1(\boldsymbol{\xi}_0, \gamma_{b-1}) & \cdots & \boldsymbol{\delta}^{3N_{\mathrm{side}}-1}(\boldsymbol{\xi}_0, \gamma_{b-1}) \\ \boldsymbol{\delta}^0(\boldsymbol{\xi}_1, \gamma_0) & \boldsymbol{\delta}^1(\boldsymbol{\xi}_1, \gamma_0) & \cdots & \boldsymbol{\delta}^{3N_{\mathrm{side}}-1}(\boldsymbol{\xi}_1, \gamma_0) \\ \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{\delta}^0(\boldsymbol{\xi}_{a-1}, \gamma_{b-1}) & \boldsymbol{\delta}^1(\boldsymbol{\xi}_{a-1}, \gamma_{b-1}) & \cdots & \boldsymbol{\delta}^{3N_{\mathrm{side}}-1}(\boldsymbol{\xi}_{a-1}, \gamma_{b-1}) \end{bmatrix} \tag{98}$$

where the rows of Z are ordered in terms of $(\boldsymbol{\xi}_p, \gamma_r)$ such that the matrix product $F = \hat{X}Z$ makes sense. Note that in computing $SOFT(\hat{X})$ the matrix Z may be pre-computed for several different values of $l_{\max}$ and stored for later use.

An algorithm for estimating the first k principal eigenimages $\tilde{\hat{U}}_k$ of $\hat{X}$ such that $\Delta\rho(\hat{X}, \tilde{\hat{U}}_k) \leq \epsilon$, where ϵ is the user specified change in energy is now presented.

SOFT Eigenspace Decomposition Algorithm

1. Form the matrix F by computing the $SOFT(\hat{X})$.

2. Form the matrix H whose columns are the ordered columns of F in descending order according to their norm.

3. Set $q = \lfloor N_{\mathrm{side}}(36N^2_{\mathrm{side}} - 1)[1 - (1/2)^{N+1}] \rfloor$, with N=0 initially.

4. Construct the matrix H_q, i.e., the matrix consisting of the first q columns of H.

5. Compute SVD$(H_q) = \tilde{\hat{U}}_q \tilde{\hat{\Sigma}}_q \tilde{\hat{V}}_q^T$. (The key observation here is that H_q contains q columns, which is considerably less than the n columns of $\hat{X}$.)

6. If $\Delta\rho(\hat{X}, \tilde{\hat{U}}_q) > \epsilon$ then set $N = N + 1$ and repeat Steps 3 through 6. Because the SVD of H_q is already available, the eigenspace can simply be updated by modifying the algorithm outlined in [39].

7. Return $\tilde{\hat{U}}_k$ such that $\Delta\rho(\hat{X}, \tilde{\hat{U}}_k) \leq \epsilon$. Note that $k \leq q$.

As with the S^2 algorithm, the above algorithm takes advantage of the fact that most SVD algorithms require mn^2 flops to compute the full SVD of $\hat{X}$; this is computationally prohibitive when n is large (which is the case for fully general pose estimation). Because the $SO(3)$ FFT is lossy, step one of the algorithm condenses the energy in $\hat{X}$ from $12bN_{\text{side}}^2$ images, where $b \approx 6N_{\text{side}}$, to $N_{\text{side}}(36N_{\text{side}}^2 - 1)$ harmonic images (roughly half) ordered in terms of the magnitude of their power spectra to form the matrix H. Furthermore, because most of the energy of $\hat{X}$ is concentrated around the lower frequency harmonics, the matrix H_q is constructed and the SVD is performed on the first $q = \lfloor \frac{N_{\text{side}}}{2}(36N_{\text{side}}^2 - 1) \rfloor$ harmonic images initially, significantly reducing the computational cost. If more harmonics are required, then half of the remaining harmonic images are concatenated to H_q and the eigenspace is updated.

It is difficult to evaluate the computational complexity of the above algorithm due to the binary split used to determine k. However, if only one iteration of steps 3-6 is performed, i.e., $N = 0$, then the complexity is on the order of mq^2 flops where $q = \lfloor \frac{N_{\text{side}}}{2}(36N_{\text{side}}^2 - 1) \rfloor$. This is the cost of computing the SVD(H_q) in step 5 of the algorithm. Similar to the algorithm developed on S^2, for all of the objects tested over 90% of the energy in $\hat{X}$ was recovered with a single itteration. (Likewise, $\Delta\rho(\hat{X}, \tilde{\hat{U}}_k) < 0.01$ when k was obtained from a single itteration.) Extensive simulation has shown that for fully general pose estimation, a subspace that can recover between 60% - 70% of the energy in $\hat{X}$ is typically more than sufficient [29].

6.5 Experimental Results

6.5.1 Test Data

The proposed algorithm detailed in Section 6.4 was tested on the objects shown in Fig. 17. The parameter $N_{\text{side}} = 5$ was used, resulting in $b = 30$ and $12bN_{\text{side}}^2 = 9000$ images per object at an angular resolution $\theta_{\text{pix}} \approx 12^\circ$. The images were then both scale and intensity normalized to create the image data matrix X. The "unbiased" image data matrix $\hat{X}$ was then constructed by subtracting the mean image from the image data matrix X. Finally, the matrix F was computed, condensing the image data set from 9000 images to 4495 harmonic images. The true SVD($\hat{X}$) was computed using MATLAB for a comparison.

6.5.2 Quality of Subspace Estimation

To validate the quality of the estimated eigenimages, the quality measures outlined in Section 2.5 were used. Fig. 24 shows all three of the quality measures averaged across all objects in Fig. 17. The top plot shows the difference in energy recovery ratio using the first 50 true eigenimages $\hat{U}_{50}$ and the first 50 estimated eigenimages $\tilde{\hat{U}}_{50}$. As can be seen from the plot, there is less than an 0.8% difference in the energy recovered using the proposed algortihm as compared to the true eigenimages. The second plot shows the residue Δ of the first 50 estimated eigenimages compared to the first 50 true eigenimages. The residue is normalized by $\sqrt{2k}$ resulting in a worst-case bound of one. The third plot shows that, based on the subspace criterion, the first 50 estimated eigenimages nearly span the same space spanned by the first 50 true eigenimages. As can be seen from all three measures, quantitatively, the estimated left singular vectors $\tilde{\hat{U}}_k$ for the proposed algorithm are very good estimates of the true left singular vectors as computed by the direct SVD.

To illustrate qualitatively the accuracy of the estimated eigenspace, the first seven eigenimages of object (8) from Fig. 17 were plotted and are presented in Fig. 25 as an example. The true eigenimages as computed by the direct SVD are shown in the top row, while the bottom row presents the eigenimages as computed by the proposed algorithm. As is apparent from a visual inspection of the figure, the first few eigenimages computed by the proposed algorithm are nearly identical to those computed by the SVD.

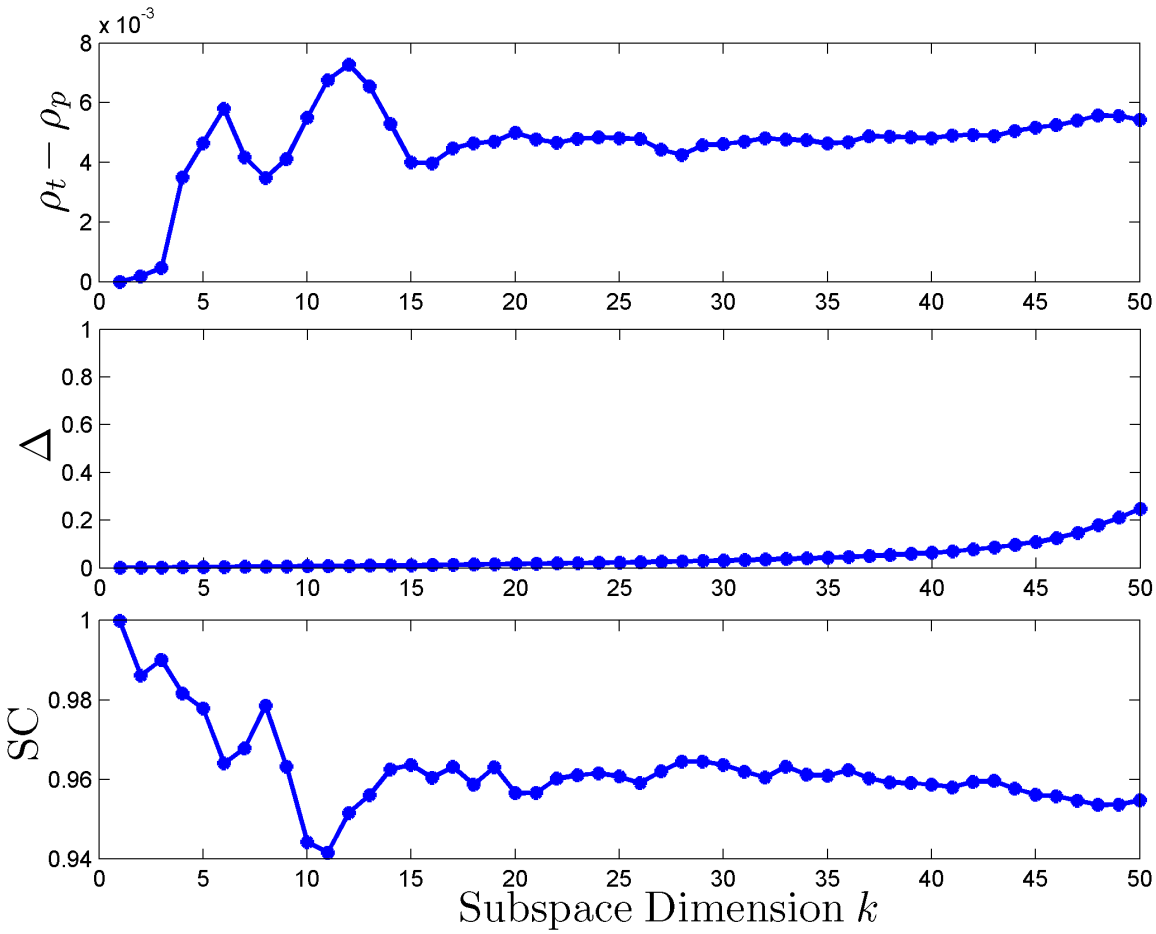

Figure 24: The quality measures outlined in Chapter 2 averaged across all objects in Fig. 17. As is apparent from the figure, the estimated left singular vectors $\tilde{\hat{U}}_k$ for the proposed algorithm are very good approximations to the true left singular vectors as computed by the direct SVD.

6.5.3 Computational Savings

Table 4 shows the required subspace dimension k, the amount of energy recovered at this subspace dimension, and the time required to estimate the first k left singular vectors $\tilde{\hat{U}}_k$ for each object depicted in Fig. 17 to meet the user specified change in energy $\epsilon = 0.01$. The user specified change in energy $\epsilon = 0.01$ implies that the k^{th} eigenimage recovers less than 1% additional energy when added to the subspace spanned by the first $k-1$ eigenimages. This result is compared to the true SVD as computed by MATLAB. As is apparent from the table, the proposed algorithm is capable of recovering nearly the same amount of energy as the direct SVD for the given subpace dimension at a significant computational savings.

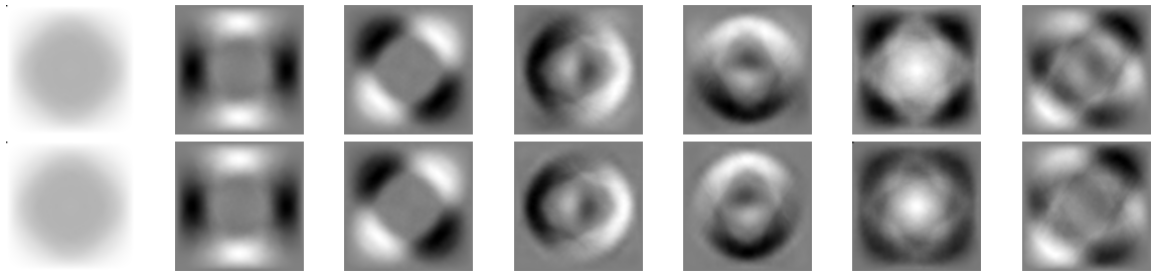

Figure 25: The first seven eigenimages of object (8) from Fig. 17. The true eigenimages as computed by the SVD using MATLAB are shown in the top row, and the eigenimages as computed by the proposed algorithm are shown in the bottom row. As is apparent from the figure, the first few eigenimages computed by the proposed algorithm are nearly identical to those computed by the SVD.

Only the first split $N = 0$ (step 3) of the algorithm was used to compute this data, resulting in an average speed-up factor of 40.

6.5.4 Accuracy of Pose Estimation

To validate that the subspace dimension k (given in Table 4) as determined by the proposed algorithm is sufficient to accurately determine the 3-D pose of arbitrary objects, each of the objects shown in Fig. 17 was sampled on a different $SO(3)$ grid resulting in 72 images per object that were not included in the original data matrix $\hat{X}$. For each of the 72 different poses, the subspace dimension k required to accurately estimate the correct pose was determined using the estimated eigenimages as computed by the proposed algorithm. Fig. 26 shows the distribution of the subspace dimension for each of the 20 object in Fig. 17. The boxes show the inner-quartile region with the horizontal bar representing the median of the data. The plus signs are the outliers in the data and represent poses that were either easier or more difficult to determine as compared to the rest of the distribution. The circles represent the subspace dimension as computed using $\Delta\rho(\hat{X}, \tilde{\hat{U}}_k) \leq 0.01$ (given in Table 4). As is apparent from the figure, with the exception of objects 9, 19, and 20, the subspace dimension as determined by computing $\Delta\rho(\hat{X}, \tilde{\hat{U}}_k)$ is sufficient to accurately determine all 72 test poses. Furthermore, only one test pose for objects 9, 19, and 20 required a subspace dimension higher than that determined by $\Delta\rho(\hat{X}, \tilde{\hat{U}}_k)$. These outliers are due to the fact that these poses lie nearly equi-distant to two adjacent poses in the original image data

Table 4: The required subspace dimension k, the amount of energy recovered at this subspace dimension, and the time required to estimate the first k left singular vectors for each object to meet the user specified change in energy $\epsilon = 0.01$. The table also outlines the average subspace dimension $\bar{k}_{\min}$ required to determine the pose of 72 test images, as well as the energy recovered at this dimension. The results are compared against the true SVD using MATLAB.

Object	k	$\bar{k}_{\min}$	Time [hours]		Energy ρ [%]		
			True	Proposed	True	Proposed	At $\bar{k}_{\min}$
1	19	5.9	3.550	0.082	69.14	66.94	37.86
2	17	5.7	3.432	0.076	68.39	68.22	47.96
3	16	6.0	3.544	0.083	64.80	63.42	43.24
4	16	4.9	3.558	0.124	54.68	54.30	29.34
5	14	6.0	3.589	0.076	63.78	62.53	43.87
6	14	6.2	3.322	0.076	76.66	76.33	60.41
7	19	5.1	2.424	0.092	60.40	60.13	33.22
8	16	5.9	3.688	0.101	45.65	39.83	24.20
9	16	5.6	3.678	0.084	60.12	58.83	41.72
10	21	5.9	3.634	0.086	58.05	59.63	33.27
11	20	5.4	3.584	0.083	64.76	64.44	31.33
12	15	5.3	3.598	0.096	57.61	56.21	39.16
13	10	5.9	3.403	0.102	50.65	42.31	29.77
14	15	5.0	3.627	0.091	64.35	64.25	43.20
15	16	5.8	3.393	0.093	66.43	64.62	43.05
16	14	5.7	3.881	0.075	70.77	70.62	56.15
17	17	5.7	3.617	0.095	60.02	59.80	38.47
18	15	5.9	3.599	0.087	68.43	65.72	47.96
19	8	5.3	3.542	0.079	71.32	66.46	54.16
20	9	6.4	4.039	0.077	64.23	57.00	51.24
Mean			3.535	0.088	63.01	61.08	50.04
Min.			2.424	0.075	45.65	39.83	47.25
Max.			4.039	0.124	76.66	76.33	51.24

matrix $\hat{X}$. Therefore, an interpolation procedure between p of the closest matching poses would resolve this issue, even with a much smaller subspace dimension k.

Table 4 also shows the average subspace dimension $\bar{k}_{\min}$ required to determine the pose of each of the 72 above mentioned test images, as well as the energy recovered at this dimension. As shown in the table, the average energy recovered at $\bar{k}_{\min}$ is 50.04% with the min and max being 47.25% and 51.24% respectively. This data supports the claim in Section 6.4 that a subspace that can recover between 60% - 70% of the energy in $\hat{X}$ is typically sufficient for accurate pose estimation.

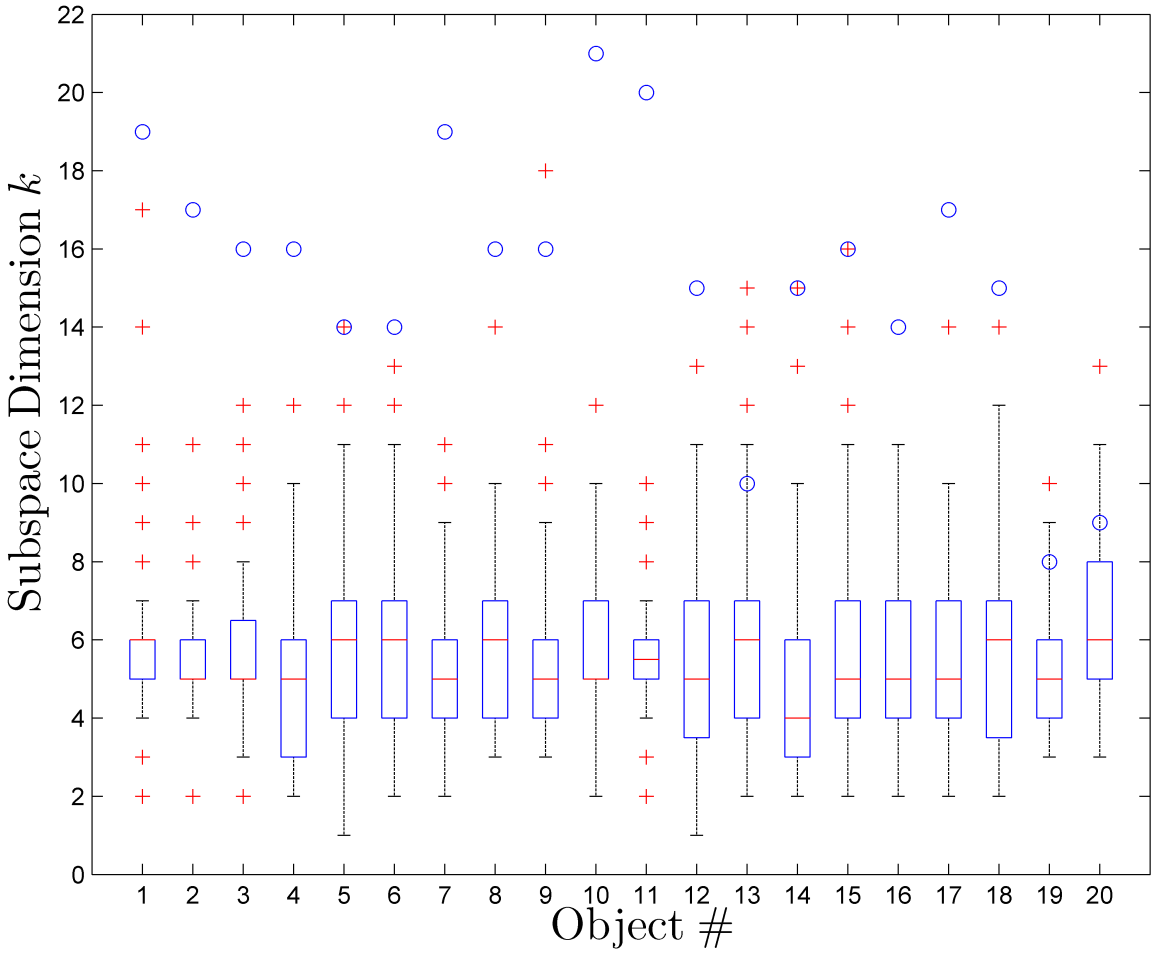

Figure 26: Box plots showing the distribution of the subspace dimension k required to determine 72 different test poses for each object in Fig. 17. The plus signs depict outliers in the data, and the circles depict the subspace dimension calculated using $\Delta\rho(\hat{X}, \tilde{\hat{U}}_k) \leq 0.01$ as shown in Table 4.

6.5.5 Maximum Energy Recovery

Based on the data provided in Table 4, and Figures 24, 25 and 26, it is obvious that the proposed algorithm is capable of estimating the required subspace dimension and left singular vectors of $\hat{X}$ at a significant computational savings for use in pose estimation. However, because the $SO(3)$ FFT is lossy, it is important to quantify the maximum amount of energy recovery possible using this approach. To do this, all $N_{\text{side}}(36N_{\text{side}}^2 - 1)$ harmonic images were used to estimate the left singular vectors of $\hat{X}$. Using these estimates, the energy recovery ratio $\rho(\hat{X}, \tilde{\hat{U}}_{N_{\text{side}}(36N_{\text{side}}^2-1)})$ was computed for each object in Fig. 17. The top plot in Fig. 27 shows the maximum amount of energy recovered per object. As seen in

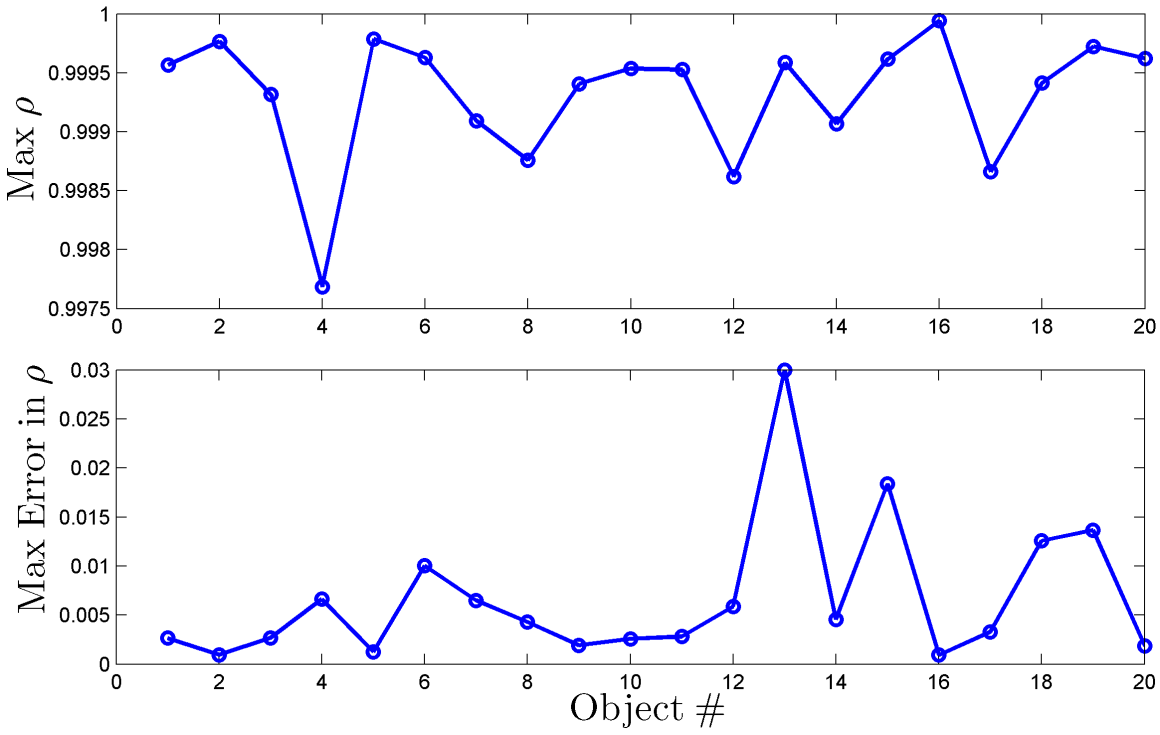

Figure 27: The maximum achievable energy recovery ratio ρ for each object in Fig. 17 [top], and the maximum difference in energy recovery between the first $N_{\text{side}}(36N_{\text{side}}^2 - 1)$ true left singular vectors of SVD($\hat{X}$) and the estimated left singular vectors $\tilde{\hat{U}}_{N_{\text{side}}(36N_{\text{side}}^2-1)}$ as computed by the proposed algorithm [bottom].

the plot, using the proposed algorithm, over 99.7% of the energy in $\hat{X}$ is recoverable for all objects. The bottom plot in Fig. 27 shows the maximum difference in energy recovery per object if only the first $N_{\text{side}}(36N_{\text{side}}^2-1)$ true left singular vectors are used to recover the same energy, i.e., the bottom plot shows $\max(\rho(\hat{X}, \hat{U}_k) - \rho(\hat{X}, \tilde{\hat{U}}_k))$ for all $k \leq N_{\text{side}}(36N_{\text{side}}^2 - 1)$ for each object. As seen in the plot, the maximum error one could expect to see is less than 3% across the entire subspace for the objects shown in Fig. 17.

CHAPTER VII

SPHERICAL HARMONICS AND ILLUMINATION VARIATION

7.1 Chapter Overview

The algorithms outlined in Chapters 3, 5, and 6 all assume the object in question is being illuminated by a single distant point light source. The remainder of this dissertation focuses on the problem of pose estimation under variations in illumination conditions when both single and multiple illumination sources are present. This chapter shows that the dimensionality of a set of images of an object under a wide range of illumination conditions and fixed pose can be significantly reduced by using the SHT developed in Chapter 5. An eigenspace decomposition algorithm is then developed to reduce the expense of computing the eigenspace decomposition of a set of images containing variations in both pose as well as illumination conditions. Section 7.2 provides an introduction to the problem of object recognition and pose estimation when variations in illumination exist. Although this dissertation does not consider the recognition phase discussed in Section 7.2, most of the related work in this area has been in the face recognition domain. Therefore, a brief discussion along with some seminal works will be presented. Section 7.3 discusses some changes in notation and function parameterization that is necessary when dealing with variations in both pose and illumination conditions. In Section 7.4, the problem of reducing the dimensionality of the image data under a fixed pose but varying illuminations conditions from a *single* illumination source is discussed. This analysis is then extended to *multiple* illumination sources in Section 7.5. An algorithm is then developed in Section 7.6 to estimate the eigenspace for a set of images that contain variation in both illumination and pose. The algorithm is tested in Section 7.7 to validate the accuracy of estimation and assess the computational

savings. Finally, in Section 7.8 a discussion is presented to outline some of the reasons the proposed algorithm is useful in terms of memory requirements and computational complexity. This section also details some alternatives to the proposed algorithm and discusses the advantages and drawbacks of each.

7.2 Introduction

It has been shown that variations among 2-D images of a 3-D object can result from four different sources, namely, photometric (illumination variability), geometric (location and pose), content variation (the object's interaction with surroundings), and non-rigid object characteristics (deformable objects) [78]. Object classification (recognition and pose estimation) under variations in illumination conditions (photometric variability) has received considerable attention in recent years. This is partially due to the fact that the same object viewed from a fixed pose can appear considerably different under different illumination conditions [78]. Furthermore, because eigenspace methods are appearance based, the appearance of the object is directly related to the robustness of the recognitions system.

This section discusses both recognition as well as pose estimation simply because the bulk of the research pertaining to variation in illumination is being conducted in the areas of facial recognition. When dealing with facial recognition problems, the subject is typically imaged from a fixed pose with variations in gesture (smiling, frowning, etc.), illumination, and in some cases, both. There are generally several subjects imaged under similar conditions with the goal being the ability to recognize test images of the subject under arbitrary conditions. Images of each subject under a variety of conditions are considered a "class", and the idea is to find a subspace to maximize the variance between classes [15].

Although the space of all possible illumination conditions is infinite, it has been shown that the illumination space can be represented by a low-dimensional linear subspace [60, 79–83]. For most 3-D objects however, variation due to a change in pose tends to be much larger than variation due to a change in illumination. Therefore, for pose estimation problems when illumination variation exists, the object itself can be treated as a separate class at each pose under a changes in illumination conditions. The general idea is then

to determine which class an input image belongs to in order to estimate its pose. This technique is detailed further in Chapter 8.

When using appearance-based methods for object recognition under photometric variability, three common approaches appear in the literature. The first approach is to capture a large set of "training" images of the object under a wide range of illumination conditions. An eigenspace decomposition is then performed and a low-dimensional subspace is chosen similar to the previous chapters. This low-dimensional subspace can then be used to classify the object in real-time [21, 24, 79, 80, 84].

If the object is largely diffuse and mostly convex (similar to human faces), a 3-D model of the object can be obtained (either by modeling it directly or using a range scanner), and the Lambertian kernel can be represented as a series of spherical harmonic coefficients. It was shown in [60, 83] that the first nine terms in the series are sufficient to capture 99% of the energy of the Lambertian kernal. Using this truncated Lambertian expansion, the principle components can be computed by evaluating the spherical harmonic basis functions at the surface normals. Unfortunatly, this technique requires the surface normals which are typically not available for most objects.

A third approach was introduced in [81, 82, 85] where the authors show that the set of images of an object under different illumination conditions forms a convex polyhedral cone (referred to as an *illumination cone*) in m-dimensional space where m is the number of pixels in the image. Furthermore, if the object is convex and Lambertian, the dimension of the illumination cone is equal to the number of distinct surface normals. In [86], Lee *et al.* extend this concept by showing that rather than attempting to capture the variation in illumination by varying the light source direction and intensity (either physically or synthetically) over all possible conditions, a set of "key" images can be captured by directing the light source from distinct locations. The authors refer to these "key" images as extreme rays and the images resulting from these extreme rays can be used as the low-dimensional subset themselves. The extreme rays discussed in [86] define the boundary of the illumination cone discussed in [81, 82, 85]. Although the idea of extreme rays is theoretically sound, how to determine the number of extreme rays required for arbitrary objects, as well as the

direction of the extreme rays is still the subject of ongoing research.

Similar to Chapters 3, 5, and 6 however, one of the key issues with using appearance-based methods for pose estimation of objects under variations in pose as well as illumination is the expense required to compute the optimal subspace for performing the estimation. This issue is especially true when variations in illumination are present because the size of the training set grows significantly when variations in illumination are considered at each pose. The remainder of this chapter presents a method to overcome this expense by using the spherical harmonic transform developed in Chapter 5. In particular, it will be shown that reducing the dimensionality of the data due to variations in illumination and a fixed pose can be efficiently done by using a truncated series of spherical harmonics. It is shown that this low-dimensional set of harmonic images is capable of recovering a significant amount of information from the original data set. Furthermore, even though the set of harmonic images are computed assuming a *single* illumination source, they are capable of recovering a significant amount of information when *multiple* illumination sources are present. Finally, an algorithm is developed to estimate the eigenspace decomposition of the entire data set (variation in illumination and pose) by applying Chang's eigenspace decomposition algorithm to the resulting set of harmonic images.

7.3 Preliminaries

Because the remainder of this dissertation deals with the problem of pose estimation under variations in illumination, some of the notation discussed in Chapter 2 is redefined to account for variations in both pose as well as illumination. Furthermore, the remainder of this dissertation only analyzes variations in pose along S^1, however the extensions to S^2 and $SO(3)$ will be obvious. To generate sets of images of objects that contain variations in illumination and pose, the objects are placed at the center of an *illumination sphere*, while the camera is moved to discrete locations on a line of constant co-latitude. At each of the discrete locations, images of the object are captured under a dense but finite set of illumination directions generated from a single distant point light source. Using this technique to capture images, the image vector can be parameterized by $\boldsymbol{f} = \boldsymbol{f}(\boldsymbol{\xi}_i, r)$ where

$r \in \{0, \ldots, a-1\}$ now represents the r^{th} pose of the object and $\boldsymbol{\xi}_i$, $i \in \{0, \ldots, b-1\}$, is the unit vector pointing at the angle of co-latitude $\beta_i \in (0, \pi)$ measured down from the upper pole, and the angle of longitude $\alpha_i \in [0, 2\pi)$, which is the parameterization of the i^{th} direction of the point light source at each pose. An example of this procedure is shown in Fig. 28 where the illumination directions are determined using the HEALPix sampling pattern [52] discussed in Chapter 4. Using this notation, the image data matrix

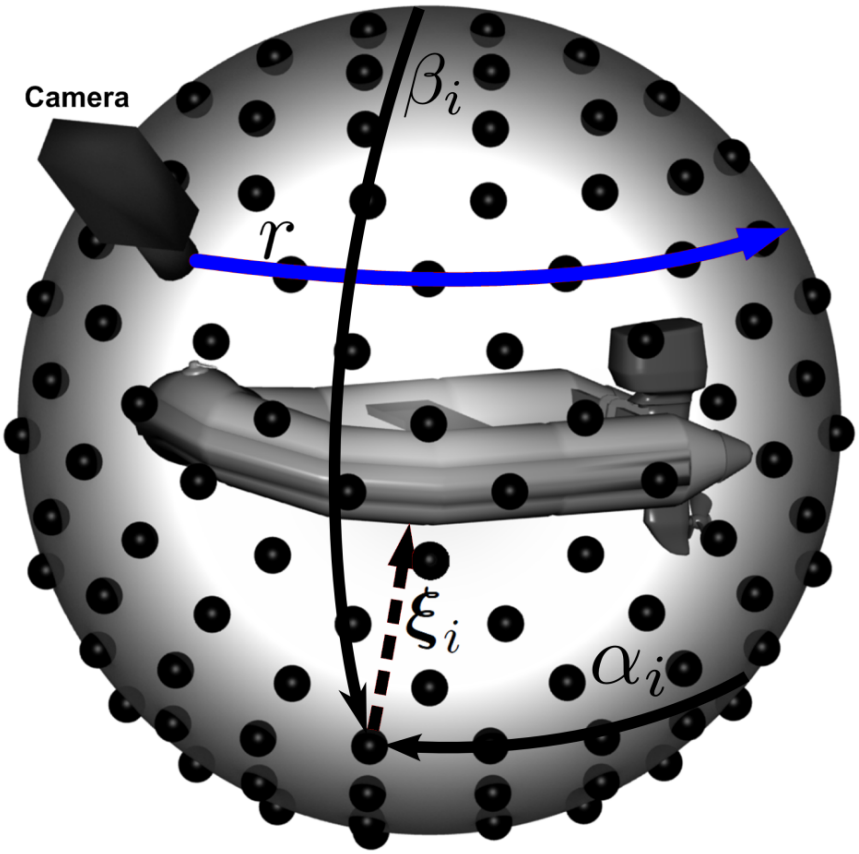

Figure 28: A graphical depiction of the proposed method of acquiring images from a dense set of illumination conditions at each pose. The object is placed at the center of the illumination sphere with the camera moving along a line of constant co-latitude. The black dots on the sphere represent different illumination conditions. As the camera moves along the line of co-latitude, an image of the object is captured under each of the distinct illumination conditions.

is constructed as

$$\begin{aligned} X &= [\boldsymbol{f}(\boldsymbol{\xi}_0,0),\boldsymbol{f}(\boldsymbol{\xi}_1,0),\ldots\boldsymbol{f}(\boldsymbol{\xi}_{b-1},0),\boldsymbol{f}(\boldsymbol{\xi}_0,1),\boldsymbol{f}(\boldsymbol{\xi}_1,1),\ldots,\boldsymbol{f}(\boldsymbol{\xi}_{b-1},1),\ldots, \\ &\quad \boldsymbol{f}(\boldsymbol{\xi}_0,a-1),\boldsymbol{f}(\boldsymbol{\xi}_1,a-1),\ldots,\boldsymbol{f}(\boldsymbol{\xi}_{b-1},a-1)], \end{aligned} \tag{99}$$

where the first b columns of X correspond to a single pose of the object under b different illumination conditions. The average image vector is then subtracted from the image data matrix X to generate the zero mean image data matrix $\hat{X}$, which has the interpretation of an "unbiased" image data matrix.

Due to the re-parameterization of $\boldsymbol{f}(\boldsymbol{\xi}_i,r)$, the spherical harmonics, and the SHT are re-indexed as

$$f(\boldsymbol{\xi}_i,r)=\sum_{p=0}^{p_{\max}}\sum_{|q|\leq p} f_{p,q}^r Y_{p,q}(\boldsymbol{\xi}_i), \tag{100}$$

with the harmonic coefficients $f_{p,q}^r$ computed using

$$f_{p,q}^r=\frac{4\pi}{b}\sum_{i=0}^{b-1} f(\boldsymbol{\xi}_i,r)Y_{p,q}(\boldsymbol{\xi}_i), \tag{101}$$

and $Y_{p,q}(\boldsymbol{\xi}_i)$ is the real-valued spherical harmonic.

7.4 Single Illumination Source

Prior to analyzing the effects of multiple illumination sources, the case where the object in question is illuminated by a single distant point light source is considered. For this development, notice that $f_{p,q}^r$ is the harmonic coefficient for a single pixel in the set of images due to a change in illumination conditions at the r^{th} pose. If all m pixels of the set of images due to a change in illumination conditions at the r^{th} pose are expanded using (101), then $\boldsymbol{f}_{p,q}^r \in \mathbb{R}^{m\times 1}$ represents a "harmonic image" of degree p and order q at pose r. The goal is to verify that for most objects, orthogonalizing the set of harmonic images provides a good approximation to the eigenimages as computed by using the SVD directly.

To illustrate this, images of each of the 20 objects shown in Fig. 17 were captured from 90 different poses and 48 different light source directions at each pose. For each of the 90 poses, the harmonic transform in (101) was used to reduce the dimensionality of the data from 48 images to 9, 16, 25, and 36 harmonic images, i.e., p =2, 3, 4, and 5, respectively.

The energy recovery ratio defined in (29) was then used to compute how much energy each of the four subspaces are capable of recovering at each pose. Fig. 29 shows the distribution of the energy recovered for each of the four subspaces across all 90 poses. The upper and lower bars on the boxes represent the maximum and minimum amount of energy recovered, respective.y. The horizontal lines in the boxes represent the median. Notice that with the exception of objects 17, 18, and 20, over 95% of the energy is recovered by the 9-dimensional

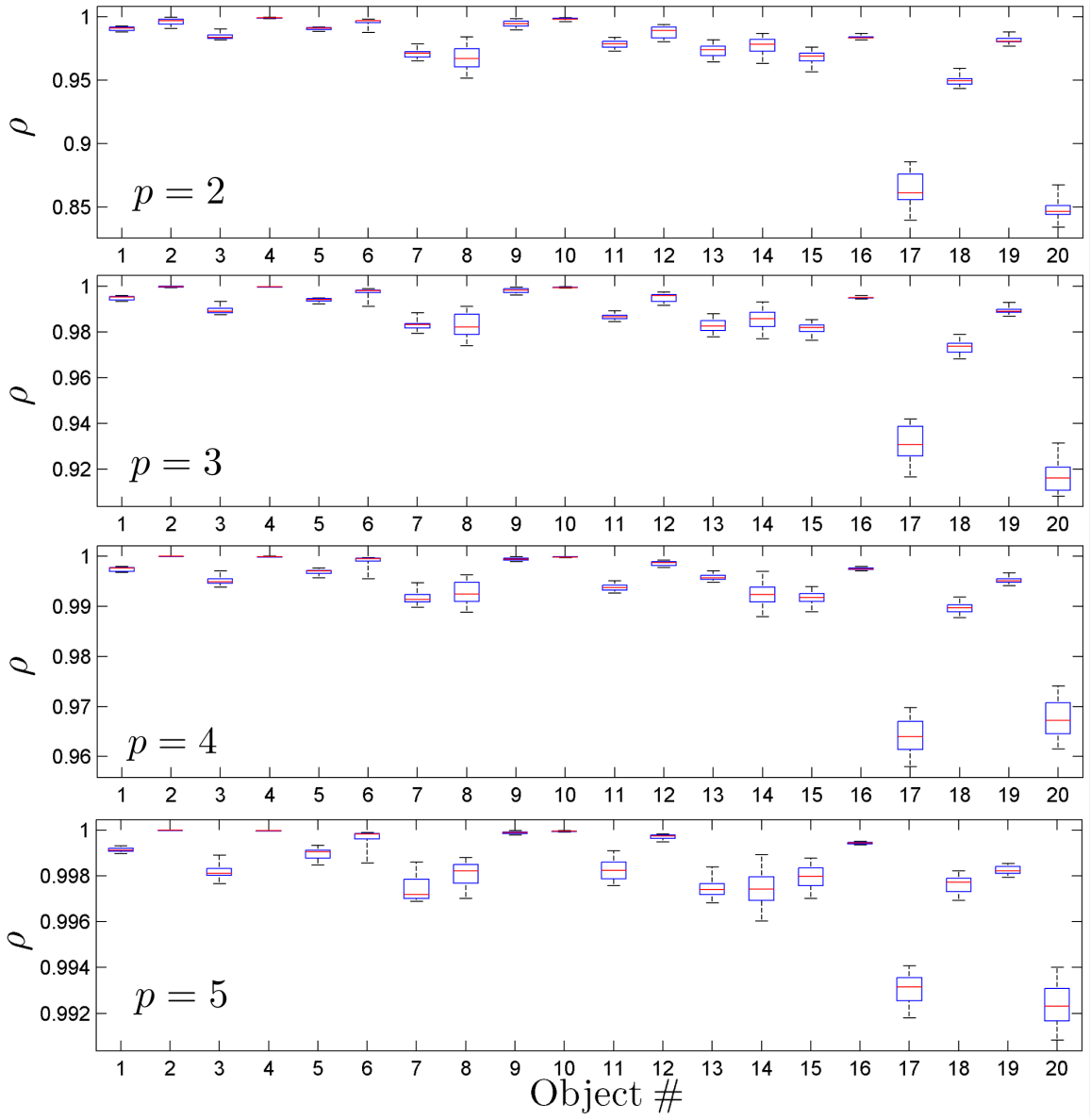

Figure 29: The minimum amount of energy recovered by 9, 16, 25, and 36 harmonic images for each object under all 90 test poses. With the exception of objects 17, 18, and 20, over 95% of the energy is recovered by the 9-D linear subspace for all 90 poses.

subspace ($p = 2$) for all 90 poses. Furthermore, adding additional harmonic images does not significantly increase the amount of energy recovered. To evaluate how well the the low-dimensional set of orthonormalized harmonic images estimate the true eigenimages as computed by the SVD, the first 9, 16, 25, and 36 true eigenimages were also computed for each of the 90 different poses. Fig. 30 shows the average difference between the energy recovered by the true eigenimages and the harmonic images across all 90 poses. As can be seen from the figure, again with the exception of objects 17, 18, and 20, there is less than 1% difference in energy recovered by the true eigenimages versus the harmonic images as computed using (101).

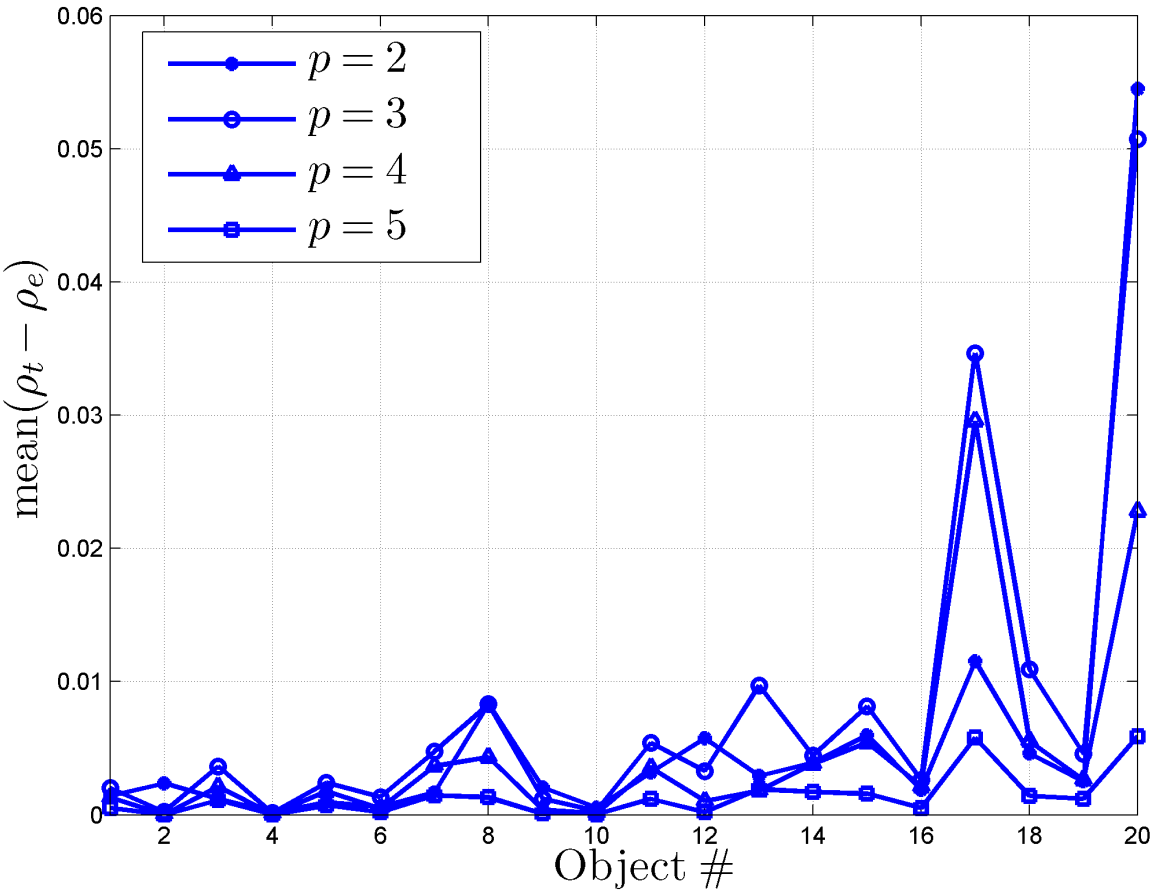

Figure 30: The difference in energy recovered by the true 9, 16, 25, and 36 dimensional subspace as computed using the SVD and the set of harmonic images as computed by the SHT for each object averaged over all 90 test poses.

Finally, the quality measures outlined in subsections 2.5.1 and 2.5.3 were used to evaluate how close the 9-dimensional subspace computed using the orthonormalized harmonic images is to the first nine eigenimages as computed by the SVD. Fig. 31 shows $1 - \mathrm{SC}$ averaged across all 90 poses for each of the 20 objects in Fig. 17. As can be seen from the figure, the harmonic images span over 85% of the same space as the first nine true eigenimages. Fig. 32 shows the normalized residue Δ averaged accross each of the 90 poses for each of the 20 objects in Fig. 17. This figure shows that for most of the objects, the first seven harmonic images are very close to the first seven true eigenimages. While in some cases the eigth and ninth harmonic image is not as close to the eigth and ninth true eigenimage, as discussed in [80], these eigenmodes typically account for sharp specular spikes (high frequency information) in the image sequence. Because the spherical harmonic transform has been truncated to order $p = 2$, the high frequency components have been effectively

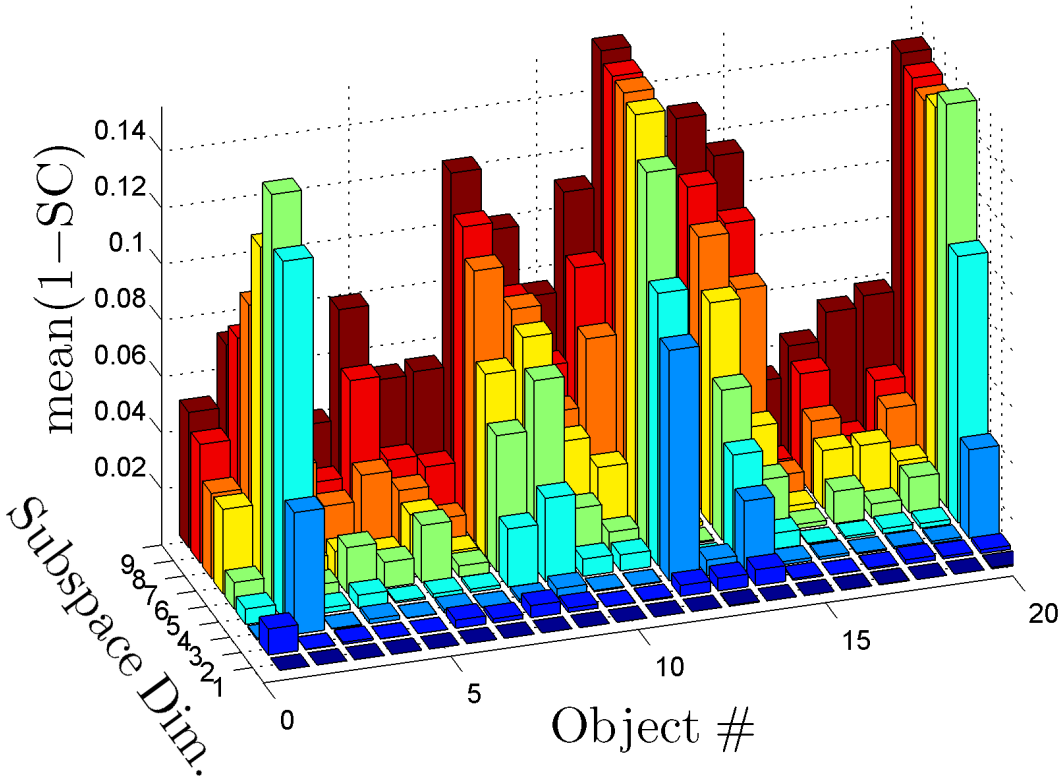

Figure 31: Subspace criterion averaged across all 90 test poses for each of the 20 objects in Fig. 17 using a 9-dimensional subspace.

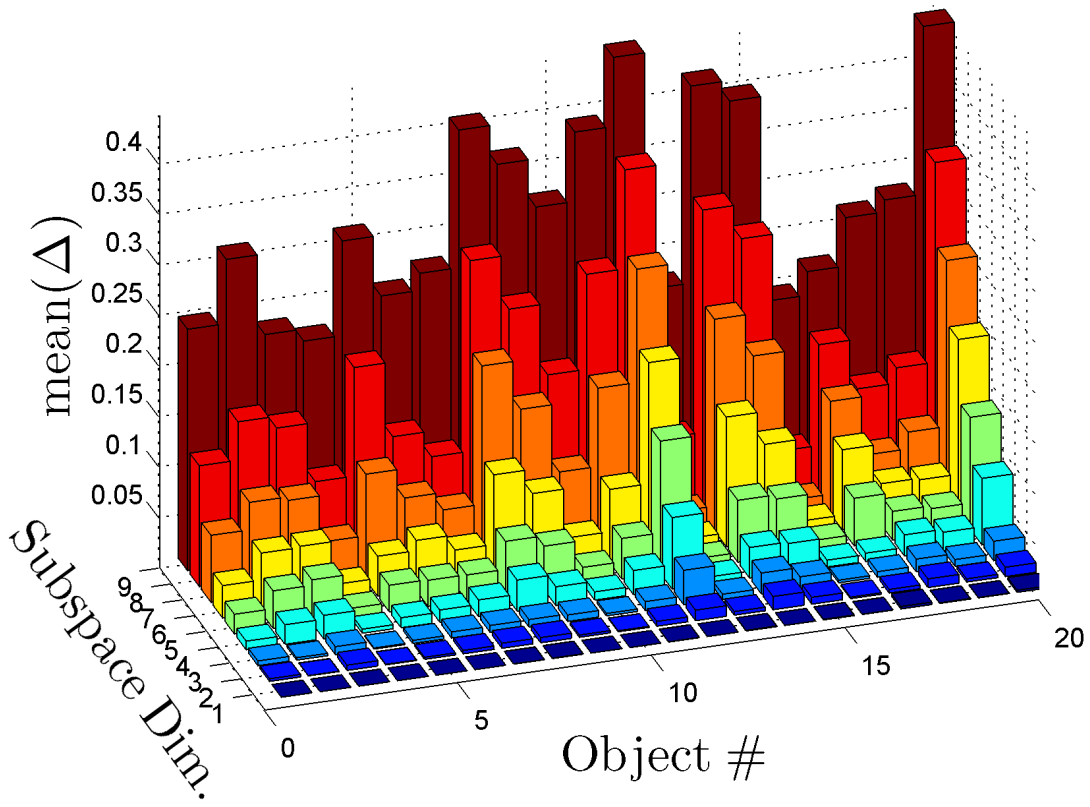

Figure 32: Normalized residue averaged across all 90 test poses for each of the 20 objects in Fig. 17 using a 9-dimensional subspace.

discarded. As such, the low-dimensional subspace can not reproduce the sharp specularities. However, the effect of this on pose estimation under variations in illumination is minimal, as will be shown in Chapter 8.

7.5 Multiple Illumination Sources

An investigation of variation due to multiple distant point light sources and fixed pose is now presented. For this evaluation, images of each of the objects in Fig. 17 were captured from 10 different poses and 48 different illumination directions at each pose. The 48 different illumination directions first consisted of a single illumination source (SS), then two illumination sources (DS), and finally three illumination sources (TS). The procedure for capturing images of the object under different illumination conditions using multiple illumination sources is as follows: First, three illumination sources were placed at random

locations around the sphere. With the camera stationary, the first source was illuminated while the other two were not, and an image of the object was captured. The second illumination source was then also illuminated, and a second image of the object was captured. Finally, the third illumination source was illuminated and the final image was captured. This process is repeated 48 times at each of the 10 poses. The sampling procedure depicted in Fig. 28 was then used to capture images of each object at each of the 10 poses and the 9-D subspace was computed using (101). Each set of harmonic images were then orthonormalized. The goal here is to investigate how effective this subspace is at recovering information from an image data set of the object from the same pose, but significantly different illumination directions/conditions. To this end, the energy recovery ratio defined

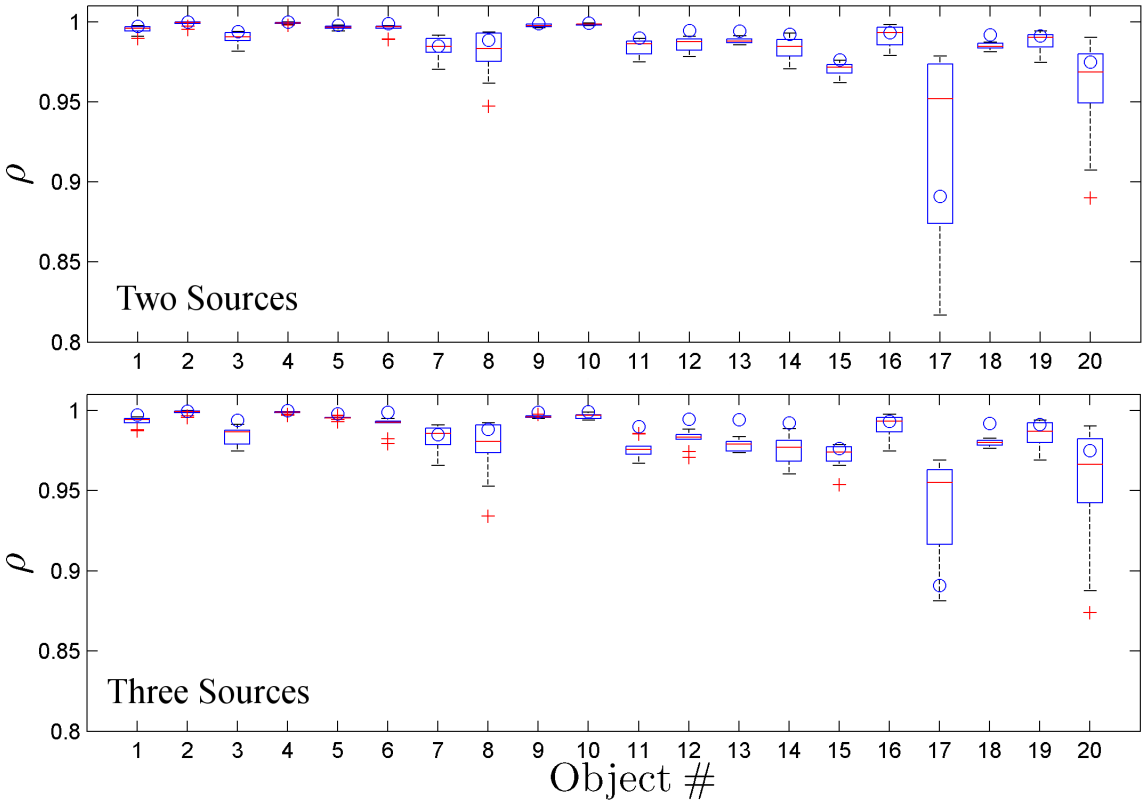

Figure 33: Distribution of the energy recovery for two [top] and three [bottom] illumination sources. The box represents the inter quartile region with the bar representing the median of the data. The plus signs represent outliers in the data, and the circles represent the median energy recovery when only a single illumination source is considered. The basis used for recovery was computed from the orthonormalized harmonic images using the first nine harmonics with a single illumination direction.

in (29) was used to compute how much of the total energy could be recovered for each of the three sets of image data (SS, DS, and TS). The results are depicted in Fig. 33. The top plot in Fig. 33 shows the distributions of the energy recovered for each of the 20 objects across all ten poses when two illumination sources are present, and the bottom plot shows the same distribution when three illumination sources are present. The horizontal bars in the boxes of Fig. 33 represent the median of the data and the circles represent the median energy recovered when only a single illumination source is present. Note that even though the median energy recovered for a single source is typically higher than that of multiple sources, for most of the objects it is only slightly higher. Furthermore, with the exception of objects 17 and 20, the distributions remain fairly tight across all 10 poses.

Some of the reasons for the drop in energy recovered when there are multiple illumination sources become clear when examining Fig. 34. The top two rows of Fig. 34 show how well the low-dimensional subspace computed from a single illumination source can reconstruct an image of object 1 in Fig. 17 under random illumination directions from three illumination

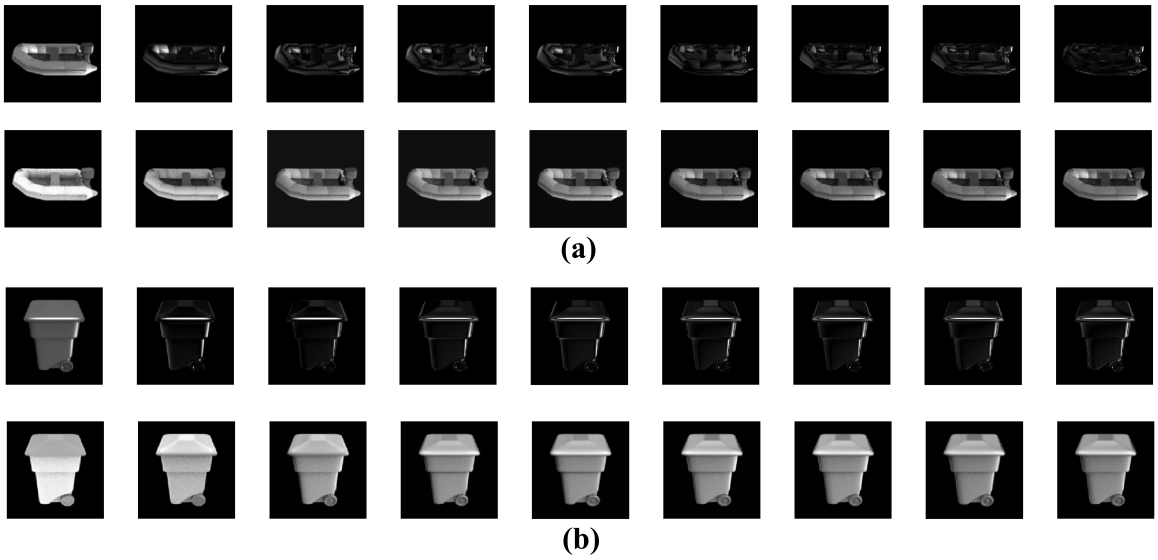

Figure 34: Reconstruction of a single image of object 1 (a) and object 20 (b) from Fig. 17 under three different illumination sources. The basis images used for reconstruction were computed using the SHT of the images generated assuming a single illumination source and the HEALPix distribution. The first image in the top row of each pair is the original image. The remaining images in the top row show the absolute difference between the reconstructed image and the image original image. The images in the second row show the reconstruction as the subspace dimension increases from left to right.

sources. The first image in the top row shows the image to be reconstructed. The image sequence in the second row shows how close the reconstruction is as additional harmonic images are used for the reconstruction. The remaining images in the top row show the absolute difference between the reconstructed image and the original image. As can be seen from the figure, the major source of reconstruction error using the 9-D subspace results from specularities in local regions of the object. Notice that for object 1 under the current viewing conditions (both viewing direction and illumination), very few local specular spikes exist. As a result, the reconstruction appears to be fairly accurate. The second sequence of images is that of object 20 from Fig. 17. The bottom two rows provide the same analysis as that given for object 1, however as can be seen from the figure, for this object the local specular regions are much larger. As a result, the low-dimensional subspace has difficulty recovering the specular spikes. Furthermore, there does not appear to be a significant improvement in reconstruction when more than a 5-dimensional subspace is used. This shows that one of the major effects of illuminating object 20 from multiple directions is that there is an increased probability that several of the local specularities will be illuminated in a single image. The effect of cast and attached shadows however is reduced due to multiple illumination sources. This result can be deduced from the fact that the probability that the shadow from a single illumination source will be illuminated by another illumination source is significantly increased. This suggests that recognition and pose estimation of largely diffuse objects with complex geometry may actually benefit when multiple illumination sources are present, however this is not analyzed in this dissertation.

To evaluate quantitatively how well the 9-D subspace is capable of reconstructing each of the 480 (10 poses and 48 illumination directions at each pose) test images for each object in Fig. 17, an alternate metric is used. If each of the row-scanned images $\boldsymbol{f}$ in $\hat{X}$ is treated as a single point in m-dimensional space, then the Euclidean norm between the true image $\boldsymbol{f}_t$ and the reconstructed image $\boldsymbol{f}_r$ provides a metric for determining how good the image reconstruction is using the low-dimensional subspace. Fig. 35 shows the distribution of $\|\boldsymbol{f}_t - \boldsymbol{f}_r\|_2$ for all 480 test images under all three illumination conditions (SS, DS, and TS). One item of interest in Fig. 35 is that the median reconstruction error is very

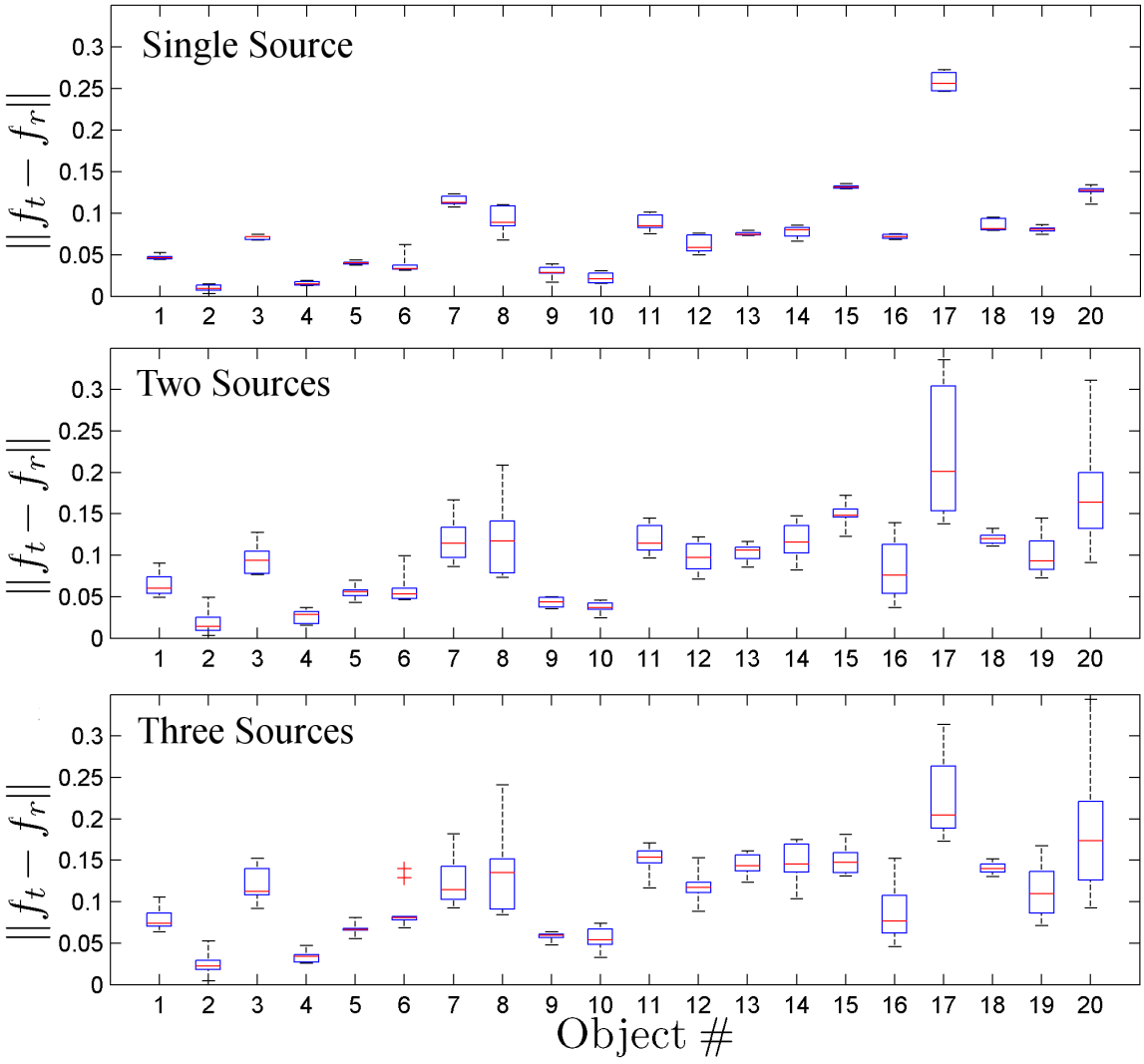

Figure 35: Distribution of $\|\boldsymbol{f}_t - \boldsymbol{f}_r\|_2$ for all 480 test images of each object using a single illumination source (SS) [top], two illumination sources (DS) [center], and three illumination sources (TS) [bottom].

similar regardless of how many illumination sources are present, however, the distributions of the reconstruction error is much higher when multiple sources are present. This again implies that there is a higher probability of illuminating a local specular region when more illumination sources are present.

7.6 Fast Eigenspace Decomposition Algorithm

The objective is to estimate the first k principal eigenimages $\tilde{\hat{U}}_k$ of $\hat{X}$ such that $\rho(\hat{X}, \tilde{\hat{U}}_k) \geq \mu$, where μ is the user specified energy recovery ratio. To this end, two observations can be

made, the first is that reducing the dimensionality of the data in the illumination dimension can be efficiently done using the analysis provided in Sections 7.4 and 7.5 resulting in a set of harmonic images $\boldsymbol{f}_{p,q}^{r}$ at each pose. Note that each harmonic image corresponds to a spherical harmonic of degree p and order q at each of the a poses. Therefore, each set of harmonic images corresponding to a given value of p and q across all r can be concatenated to form the matrix

$$\hat{X}_{p,q} = [\boldsymbol{f}_{p,q}^{0}, \boldsymbol{f}_{p,q}^{1}, \cdots, \boldsymbol{f}_{p,q}^{a-1}]. \tag{102}$$

Furthermore, because the harmonic expansion is truncated, there will be nine such matrices in total, each of size $\mathbb{R}^{m\times a}$.

Each of the nine matrices in (102) now only contain variations due to a change in pose for a given spherical harmonic coefficient, and thus are correlated in a single dimension. Therefore, the second observation that can be made is that the dimensionality of the data in the pose dimension can be reduced by applying the results observed by Chang *et al.* (refer to Chapter 3) to each of the nine matrices $\hat{X}_{p,q}$ [30]. In other words, it can be assumed that the right singular vectors of $\hat{X}_{p,q}$ are well-approximated by a few low-frequency Fourier harmonics, and the FFT can be used to determine $\rho(\hat{X}_{p,q}^{T}, H_{j_i}) \geq \mu_t$ for each of the nine p, q combinations, where μ_t is a user specified value for the energy recovery along the pose dimension for each of the nine harmonics. Notice that j_i, $i = 1, 2, \ldots, 9$, corresponds to the number of Fourier harmonics required for the i^{th} (p, q) combination to achieve the user specified energy recovery ratio μ_t. Let $Z_{j_i}^{p,q}$ denote the matrix $\hat{X}_{p,q}H_{j_i} \in \mathbb{R}^{m\times j_i}$ for each (p, q) combination, and construct the reduced order matrix

$$\bar{X} = [Z_{j_1}^{0,0}, Z_{j_2}^{1,-1}, Z_{j_3}^{1,0}, Z_{j_4}^{1,1}, Z_{j_5}^{2,-2}, \cdots, Z_{j_9}^{2,2}], \tag{103}$$

that effectively recombines the image data due to variation in both illumination and pose into a single matrix. Note that the matrix $\bar{X}$ has considerably fewer columns than that of $\hat{X}$. Furthermore, because most SVD algorithms require $\mathcal{O}(mn^2)$ flops, computing the dominant left singular vectors $\tilde{\hat{U}}_k$ of $\bar{X}$ by means of the SVD results in excellent estimates of $\hat{U}_k$ at a significant computational savings. The entire algorithm is summarized as follows:

Eigenspace Decomposition Algorithm Summary

1. Use the SHT to compute the matrices
   $P_r = [\boldsymbol{f}^r_{0,0}, \boldsymbol{f}^r_{1,-1}, \boldsymbol{f}^r_{1,0}, \boldsymbol{f}^r_{1,1}, \boldsymbol{f}^r_{2,-2}, \cdots, \boldsymbol{f}^r_{2,2}]$ for all r.

2. Construct the matrices $\hat{X}_{p,q}$ by concatenating each of the harmonic images $\boldsymbol{f}^r_{p,q}$ in P_r for each r as shown in (102).

3. For each of the nine matrices $\hat{X}_{p,q}$, determine the smallest number j_i such that $\rho(\hat{X}^T_{p,q}, H_{j_i}) \geq \mu_t$, where μ_t is the user specified energy recovery ratio in the pose dimension, and $i = 1, 2, \ldots, 9$ corresponds to the i^{th} matrix $\hat{X}_{p,q}$.

4. Let $Z^{p,q}_{j_i}$ denote the matrix $\hat{X}_{p,q}H_{j_i}$ and construct the matrix $\bar{X}$ defined in (103). Note that the matrices $Z^{p,q}_{j_i}$ can be efficiently computed using the FFT.

5. Compute the SVD of $\bar{X} = \tilde{\hat{U}}\tilde{\hat{\Sigma}}\tilde{\hat{V}}$.

6. Return $\rho(\hat{X}, \tilde{\hat{U}}_k) \geq \mu$. Where μ is the user specified energy recovery ratio.

7.7 Experimental Results

7.7.1 Test Data

The proposed algorithm detailed in Section 7.6 was tested on each of the objects in Fig. 17. Recall that each of the objects was sampled at a resolution of 128×128 from 90 different poses under 48 different light source locations at each pose. To accurately represent real objects using CAD models, the reflectance model used accounts for material properties such as surface roughness and surface hardness, and incorporates a mix of diffuse and specular reflection using the Cook-Torrance reflectance model [87]. The mean image was then subtracted to construct the image data matrix $\hat{X}$. The parameters used in the algorithm were $\mu_t = 0.95$ for the pose reduction and $\mu = 0.8$ for the total energy recovered. The true SVD of the image data matrix $\hat{X}$ was also computed using MATLAB for comparison. The quality measures outlined in Section 2.5 were used to evaluate the accuracy of the estimated subspace.

7.7.2 Performance and Computational Savings

Table 5 shows the required subspace dimension k and the time required to estimate the first k left singular vectors $\tilde{\hat{U}}_k$ for each object in Fig. 17 to meet the user specified energy recovery ratio $\mu = 0.8$. This result is compared to the true SVD as computed by MATLAB. As is apparent from the table, using the proposed algorithm, the left singular vectors $\tilde{\hat{U}}_k$ are very good estimates of $\hat{U}_k$ at a significant computational savings. Table 5 also shows the column dimension of $\bar{X}$ in step 5 of the proposed algorithm. Note that for the current test data, the number of columns in $\hat{X}$ is 4320, whereas for all 20 objects in Fig. 17, the number of columns in $\bar{X}$ never exceeds 576, thus resulting in significant computational savings.

Table 5: Subspace dimension k and the time required to estimate the first k left singular vectors for each object to meet the user specified energy recovery ratio $\mu = 0.8$. The results are compared against the true SVD using MATLAB. The table also shows the column dimension of $\bar{X}$.

Object	Dim. k		Time [min.]		Col. Dim. of $\bar{X}$
	True	Proposed	True	Proposed	
1	17	17	31.3	0.111	378
2	9	9	25.5	0.070	162
3	13	13	32.3	0.116	379
4	15	15	30.0	0.137	474
5	10	10	31.6	0.076	229
6	14	15	31.0	0.181	576
7	16	17	27.9	0.099	239
8	31	31	31.6	0.152	446
9	19	19	30.8	0.162	502
10	14	14	31.6	0.154	448
11	22	22	31.7	0.117	356
12	20	20	31.8	0.188	561
13	8	8	21.1	0.114	254
14	12	12	30.8	0.107	270
15	23	23	30.8	0.153	472
16	27	27	21.3	0.109	249
17	196	217	22.9	0.183	552
18	20	20	15.5	0.093	173
19	25	25	21.5	0.152	439
20	33	46	21.4	0.099	209
Mean			27.6	0.129	368
Min.			15.5	0.070	162
Max.			32.3	0.188	576

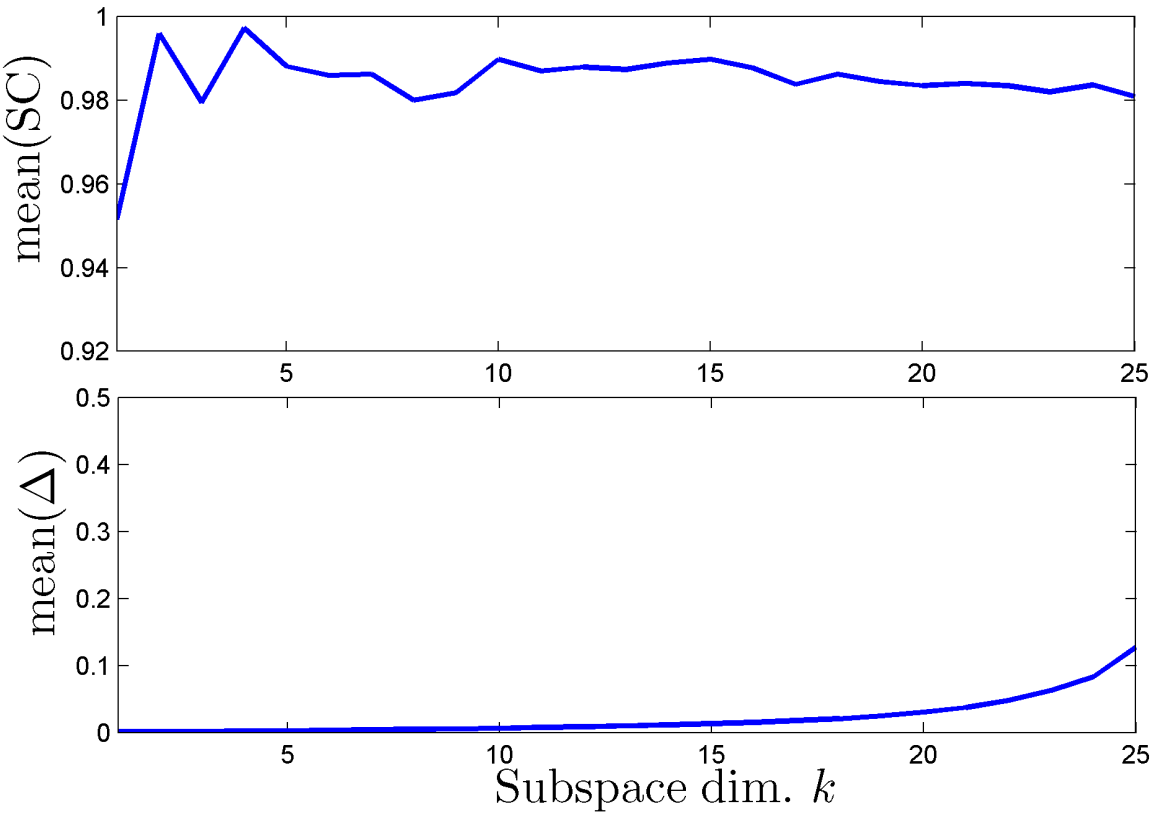

Figure 36: The subspace criterion SC [top] and the normalized residue Δ [bottom] as a function of the subspace dimension k averaged across all objects in Fig. 17.

Fig. 36 shows the subspace criterion SC [top] and the normalized residue Δ [bottom] as a function of the subspace dimension k averaged across all objects in Fig. 17. As can be seen from the figure, the estimated left singular vectors $\tilde{\hat{U}}_k$ computed by the proposed algorithm are very good approximations to the true left singular vectors as computed by the direct SVD in terms of spanning the same subspace as $\hat{X}$. Fig. 37 shows the difference in energy recovered by the true SVD and the proposed algorithm for all objects in Fig. 17. As can be seen from the figure, with the exception of objects 17 and 20, there is less than a 0.25% difference in how much energy the proposed algorithm is capable of recovering compared to the true SVD. The subspace dimension used for the energy calculation is outlined in column 2 of Table 5. As is apparent from Table 5, and Figs. 36 and 37, the proposed algorithm is capable of accuratly estimating the left singular vectors $\tilde{\hat{U}}_k$ at a significant computational savings.

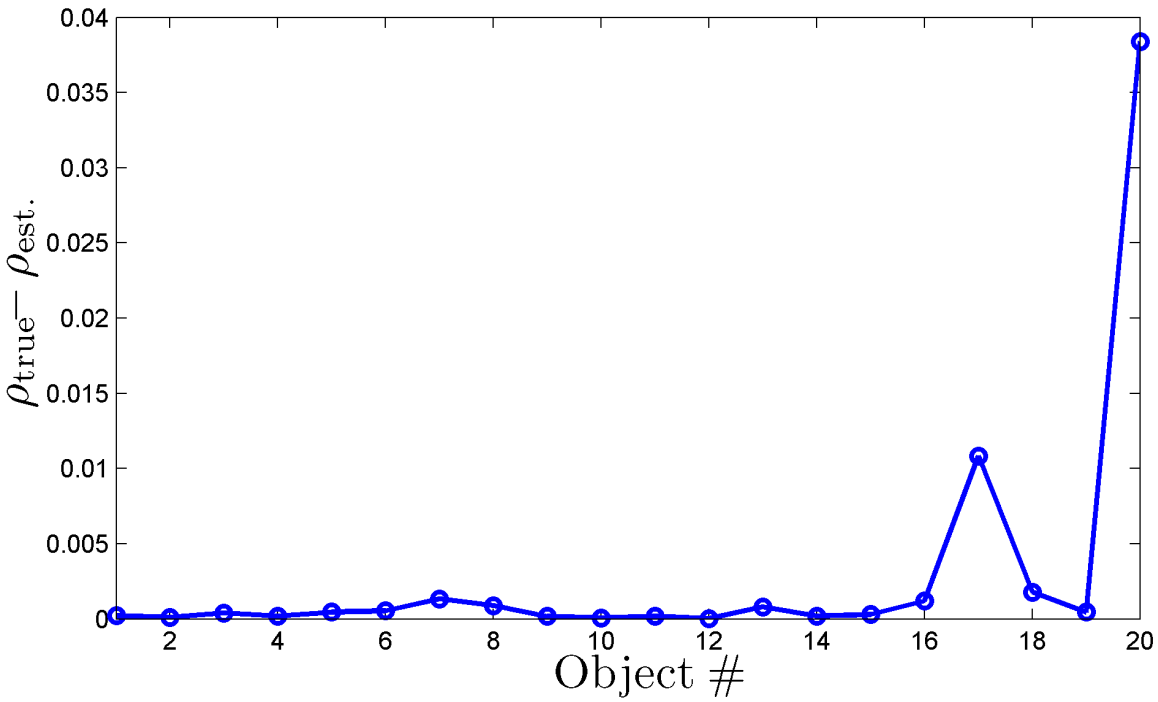

Figure 37: The difference in energy recovered by the true SVD and the proposed algorithm for all objects in Fig. 17. The subspace dimension used for the calculation is listed in column 2 of Table 5

7.8 Discussion

In this subsection, an alternative solution to the current eigenspace decomposition algorithm is discussed, along with some of the advantages and disadvantages associated with it. Before this discussion however, note that reducing the dimension of the image data matrix $\hat{X}$, and as such reducing the computational expense associated with computing the SVD, is one key advantage of the proposed algorithm. Another advantage is the reduction in memory requirements. The naive approach to computing the eigenspace decomposition of $\hat{X}$ requires that the $m \times n$ matrix be stored in memory before it can be operated on. For the data provided in Section 7.7, this requires a data structure of size $128^2 \times 4320$. If each element in the data structure is represented by 8-bits, the memory requirements for storage exceed 500 Mb. This does not account for memory required to compute on the matrix, i.e., memory for the SVD, swap space, etc. Furthermore, if the object requires a more dense sampling in the illumination dimension, the storage requirements rapidly become infeasible. The proposed algorithm significantly reduces the storage requirements because a reduction in

the illumination dimension can be computed independently for each pose and using the SHT only requires individual images of the object under distinct illumination conditions. Therefore, the storage requirements for each pose are reduced to a data structure of size $128^2 \times 9$. The reduction in the pose dimension depends on object geometry and accuracy, however the total dimension of $\bar{X}$ is significantly less than that of $\hat{X}$ as illustrated in Table 5.

It is apparent that first reducing the dimensionality due to variation in illumination and subsequently reducing the dimensionality of the data due to a change in pose is advantageous for computational reasons, as well as storage requirements. Therefore, several alternatives to the proposed algorithm could also be implemented, each of which has advantages as well as drawbacks. One such alternative would be to perform an SHT at each pose, and then use the SVD on a subset of the harmonic images to reduce the dimensionality of the data due to a change in illumination conditions. The results observed by Chang *et al.* could then be applied to reduce the dimensionality of the data due to a change in pose. The advantages of this are that more energy can be captured due to a change in illumination before reducing the dimensionality of the data in the pose dimension, and the computational savings would still be significant as compared to the naive approach. The major drawbacks to using this alternative are two fold, the first is the increased computation time in computing the SVD vs. the SHT. The SHT can be computed in $\mathcal{O}(mn\log^2(n))$ operations whereas the SVD requires $\mathcal{O}(mn^2)$ operations. The second is that there is no guarantee that the ordering of the left singular vectors computed to reduce the dimensionality due to variation in illumination at each pose will be the same from pose to pose. Therefore, for each pose, the ordering of the left singular vectors would need to be completed before reducing the dimensionality in the pose dimension. The only way the author believes this could be done is by ordering them using a brute force approach which would require a significant increase in computational expense. Using the SHT on the other hand, the harmonic images at each pose correspond to a given spherical harmonic $Y_{p,q}$ and as such, no ordering is required. Therefore, the ordering remains consistent throughout the entire pose dimension. Other variants of this could also be applied, however, they all have similar advantages and disadvantages.

CHAPTER VIII

POSE ESTIMATION AND MANIFOLD ANALYSIS

8.1 Chapter Overview

This chapter outlines how the estimated eigenimages $\tilde{\hat{U}}_k$ can be used for illumination invariant pose estimation. The chapter begins with an introduction to current techniques used when dealing with manifold analysis and closest point searches in high-dimensional spaces in Section 8.2. An analysis of eigenspace manifolds when variations in pose along S^1 and illumination exist is presented in Section 8.3. In this section, an alternative to the technique discussed in subsection 2.3.3 is developed based on the concept of an illumination manifold and eigenspace partitioning. An evaluation of the technique developed in Section 8.3 is presented in Section 8.4. The evaluation is based on estimating the pose of the objects in Fig. 17 from several test images that contain variations in pose as well as illumination from single and multiple illumination sources. The chapter concludes with a brief discussion on a variation of the technique developed in Section 8.3 when the illumination directions are known *a priori*.

8.2 Introduction

The algorithm developed in Chapter 7 provides an efficient method to compute the principle eigenimages of a set of images that contain variations in both pose and illumination. Once the principle eigenimages have been computed, the question of how to accurately and efficiently estimate the pose of the object arises.

The naive approach to estimate the pose of the object is to use the method outlined in subsection 2.3.3 and search the eigenspace manifold using an exhaustive search. However searching the entire eigenspace becomes prohibitively expensive for real-time applications,

particularly when the number of images and/or the dimension of the eigenspace increases from S^1 to S^2, and finally, $SO(3)$. Other methods include space partitioning algorithms such as R-tree, k-d tree, and Voronoi polygons [88–91], hashing techniques [92], and random search methods [93]. The drawback of these techniques is that they are either inaccurate, require a large amount of storage space, or their computational complexity becomes large as the dimension of the space increases. In [94], Nene and Nayar propose a method to search for nearest neighbor points in high dimensions using a space partitioning algorithm that is based on the projection search paradigm. The algorithm consists of generating hyperplanes in the eigenspace that ultimately "box-in" the point of interest with a hypercube, each side of the hypercube having length 2ϵ. An exhaustive search is then performed within this hypercube to find the closest matching point. The authors of [94] show that this algorithm is computationally less expensive than previous search techniques, however its complexity is still a function of the dimension of the space.

In [95], a fundamentally different approach for estimating the pose of 3-D objects using eigenspace methods was presented when the object was correlated on S^1 with constant ambient illumination. It was shown that rather than choosing the principle eigenimages as the optimal set of basis vectors, a linear combination of this basis could be computed so that the resulting eigenspace manifold has a desirable geometric structure. This geometric structure allows for the pose of the object to be computed using simple calculations rather than searching the eigenspace. The advantage of this technique stems from the fact that, because it requires no search of the eigenspace, its computational expense is constant. Unfortunately, the robustness of this technique is extremely sensitive to small perturbations from the original training images. This drawback results in poor estimation accuracy when variations in both pose and illumination exist.

Another class of techniques for dealing with the complexity of nonlinear manifolds in a high-dimensional space are referred to as manifold learning. Manifold learning is based on nonlinear dimensionality reduction and attempts to preserve local regions of the manifold geometry. The most widely used manifold learning techniques are locally linear embedding (LLE), and isometric mapping (ISOMAP) [96–98]. Both techniques are based on evaluating

the structure of k nearest neighbors and embedding that structure into a low-dimensional manifold. Their differences lie in how the "evaluation" of the k nearest neighbors is performed and how the embedding occurs. The results of both is a low-dimensional representation of the high-dimensional manifold that maintains some of the local and global characteristics. The major pitfall with both techniques for the application of pose estimation however is that there is no efficient method to evaluate new incoming data.

In the remainder of this chapter it will be shown that when dealing with the problem of pose estimation under a wide range of illumination conditions, the entire eigenspace may not need to be searched. It will be shown that for most objects, variations due to a change in pose tend to be much larger than those due to a change in illumination. Based on this observation, a technique is presented to partition the eigenspace into smaller search regions, thus reducing the computational expense. This analysis also provides some insight into the structure of eigenspace manifolds when variations in pose and illumination exist [30].

8.3 Analysis

In this section, a discussion of how the estimated eigenimages $\tilde{\hat{U}}_k$ can be used for illumination invariant pose estimation is presented. Table 6 illustrates some definitions that will be relevant for this discussion.

Pose estimation using eigenspace decomposition consists of first computing the eigenspace using the algorithm developed in Chapter 7, and then projecting the training images into the eigenspace by computing $\mathcal{M} = \tilde{\hat{U}}_k^T \hat{X}$. The matrix $\mathcal{M}$ is a discrete approximation to a manifold embedded in k-dimensional space consisting of n points. Because variations due to a change in pose are larger than variations due to a change in illumination for most objects, there are typically a "clusters" of points with each cluster containing b points. Each of the a clusters, denoted $\mathcal{I}_r$, is a discrete approximation to the illumination manifold at pose r. Note that $\mathcal{I}_r$ is a set of b points corresponding to the r^{th} pose, each point having dimension k. An example of this is depicted in Fig. 38. The figure shows the projection of the image data for object 1 in Fig. 17 onto the first three estimated eigenimages, i.e., the figure shows $\mathcal{M} = \tilde{\hat{U}}_3^T \hat{X}$. Each of the polygonal meshes represents an "illumination manifold" $\mathcal{I}_r$ at

Table 6: Table of manifold definitions.

Symbol	Construction	Description
n	Definition	Total number of images.
a	Definition	Number of poses.
b	Definition	Number of illumination directions.
k	Definition	Subspace dimension.
r	$r \in \{0, 1, \cdots, a-1\}$	r^{th} pose of the object.
$\mathcal{M} \in \mathbb{R}^{k \times n}$	$\mathcal{M} = \tilde{\hat{U}}_k^T \hat{X}$	Approximation to the manifold embedded in k-dimensional space corresponding to an object under all illumination directions and all changes in pose.
$\mathcal{I}_r \in \mathbb{R}^{k \times b}$	$\mathcal{I}_r = \tilde{\hat{U}}_k^T \hat{X}_r$	Approximation to the illumination manifold at pose r.[1]
$\mathbf{c}_r \in \mathbb{R}^{k \times 1}$	$\frac{1}{b} \sum_{i=1}^{b} \mathcal{I}_r^{(i)}$	Center of mass of illumination manifold $\mathcal{I}_r$.[2]
$\mathcal{C} \in \mathbb{R}^{k \times a}$	$\mathcal{C} = [\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_a]$	Approximation to the manifold through the center of mass of $\mathcal{I}_r$ for all r.
$\mathbf{p}_r^{r+1} \in \mathbb{R}^{b \times 1}$	$\mathbf{p}_r^{r+1} = \mathcal{I}_r^T(\mathbf{c}_{r+1} - \mathbf{c}_r)$	Orthogonal projection of $\mathcal{I}_r$ onto the linear approximation to the manifold generated from $\mathbf{c}_r$ to $\mathbf{c}_{r+1}$.
$\mathbf{p}_r^{r-1} \in \mathbb{R}^{b \times 1}$	$\mathbf{p}_r^{r-1} = \mathcal{I}_r^T(\mathbf{c}_{r-1} - \mathbf{c}_r)$	Orthogonal projection of $\mathcal{I}_r$ onto the linear approximation to the manifold generated from $\mathbf{c}_r$ to $\mathbf{c}_{r-1}$.
$\Delta\mathcal{C} \in \mathbb{R}^{1 \times a}$	See 3	Distance between each point in $\mathcal{C}$.

[1] Recall that $\hat{X}_r$ correspond to the matrix containing images of the object under all illumination directions at pose r

[2] (i) corresponds to the i^{th} column of $\mathcal{I}_r$

[3] $\Delta\mathcal{C} = [\|\mathbf{c}_2 - \mathbf{c}_1\|, \|\mathbf{c}_3 - \mathbf{c}_2\|, \cdots, \|\mathbf{c}_a - \mathbf{c}_{a-1}\|, \|\mathbf{c}_1 - \mathbf{c}_a\|]$ where the last term results from the fact that $\mathcal{C}$ is closed

each of the a poses. To estimate the pose of the object in question, an input image $\boldsymbol{f}_i$ is projected into the eigenspace using $\mathcal{P} = \tilde{\hat{U}}_k^T \boldsymbol{f}_i$, where $\mathcal{P}$ is a k-dimensional point in the eigenspace. The goal is then to determine which of the a clusters is closest to $\mathcal{P}$ (in the Euclidean sense).

Notice that for most poses in Fig. 38, the illumination manifolds do not overlap. This implies that for object 1, variations due to a change in illumination are significantly less than, or orthogonal to, those due to a change in pose. The line connecting each of the a illumination manifolds in Fig. 38 is referred to as the manifold centroid. The manifold centroid, denoted $\mathcal{C}$, is computed by calculating the center of mass of each cluster, i.e.

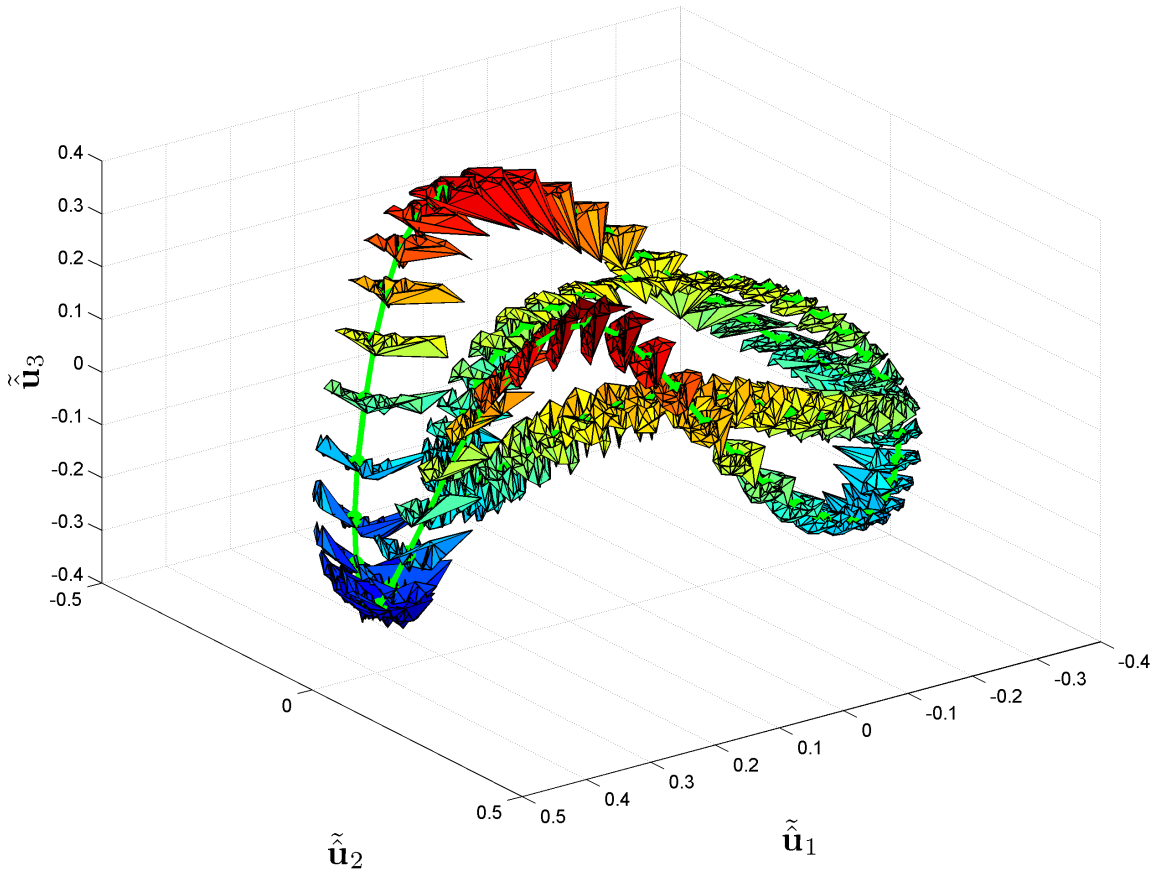

Figure 38: The projection of the image data for object 1 in Fig. 17 onto the first three estimated eigenimages.

$\mathcal{C} = [\mathbf{c}_0, \mathbf{c}_2, \ldots, \mathbf{c}_{a-1}]$, where

$$\mathbf{c}_r = \frac{1}{b} \sum_{i=1}^{b} \mathcal{I}_r^{(i)} \tag{104}$$

and (i) corresponds to the i^{th} column of $\mathcal{I}_r$. Note that $\mathcal{C}$ has significantly fewer points than $\mathcal{M}$. The general idea is then to determine which point in $\mathcal{C}$ is closest to $\mathcal{P}$. However, because $\mathcal{C}$ is computed using the centroids of the illumination manifolds, simply using the point in $\mathcal{C}$ closest to $\mathcal{P}$ to estimate the pose may not be sufficient. The reasons for this become clear when examining Fig. 39 which is a graphical depiction of three points on the manifold centroid $\mathcal{C}$, two illumination manifolds corresponding to pose r and $r+1$, and the projection of an input image $\mathcal{P}$. Notice that the point $\mathcal{P}$ is actually closer to $\mathbf{c}_r$ than $\mathbf{c}_{r+1}$ even though the point is clearly closer to illumination manifold $\mathcal{I}_{r+1}$. Therefore, to determine which illumination manifold is closest to $\mathcal{P}$, the two points in $\mathcal{C}$ that are closest

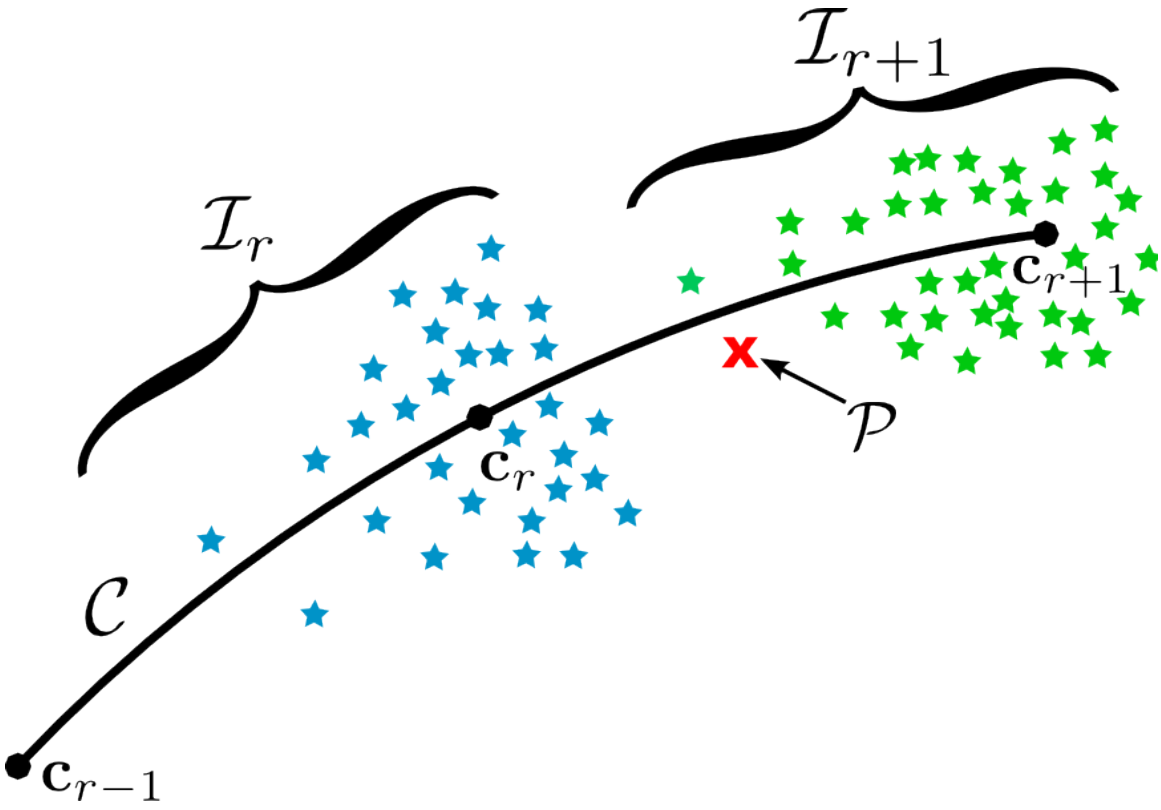

Figure 39: A graphical depiction of three points on the manifold centroid $\mathcal{C}$, two illumination manifolds corresponding to pose r and $r+1$, and the projection of an input image $\mathcal{P}$. Notice that the point $\mathcal{P}$ is actually closer to $\mathbf{c}_r$ than $\mathbf{c}_{r+1}$ even though the point clearly belongs to illumination manifold $\mathcal{I}_{r+1}$.

to $\mathcal{P}$ are used and the illumination manifolds corresponding to each are searched. For the example in Fig. 39, this would imply that both $\mathcal{I}_r$ and $\mathcal{I}_{r+1}$ would be searched to determine which illumination manifold is closest to $\mathcal{P}$.

Although partitioning the eigenspace search in this way is typically sufficient for accurate pose estimation of most objects (as will be seen in the next section), the illumination manifolds for some objects may intersect from one pose to the next. This intersection may cause issues because it is unknown *a priori* where the point $\mathcal{P}$ will lie in the eigenspace. If the point lies withing the intersection of two (or more) illumination manifolds, determining an estimate of the pose is an ill-conditioned problem. An example of this behavior is illustrated in Fig. 40 which shows three illumination manifolds $\mathcal{I}_r$, $\mathcal{I}_{r+1}$, and $\mathcal{I}_{r+2}$. The projection of an input image $\mathcal{P}$ is also shown in the figure. Notice that all three of the illumination manifolds significantly overlap, as a result, it is very difficult to determine which of the

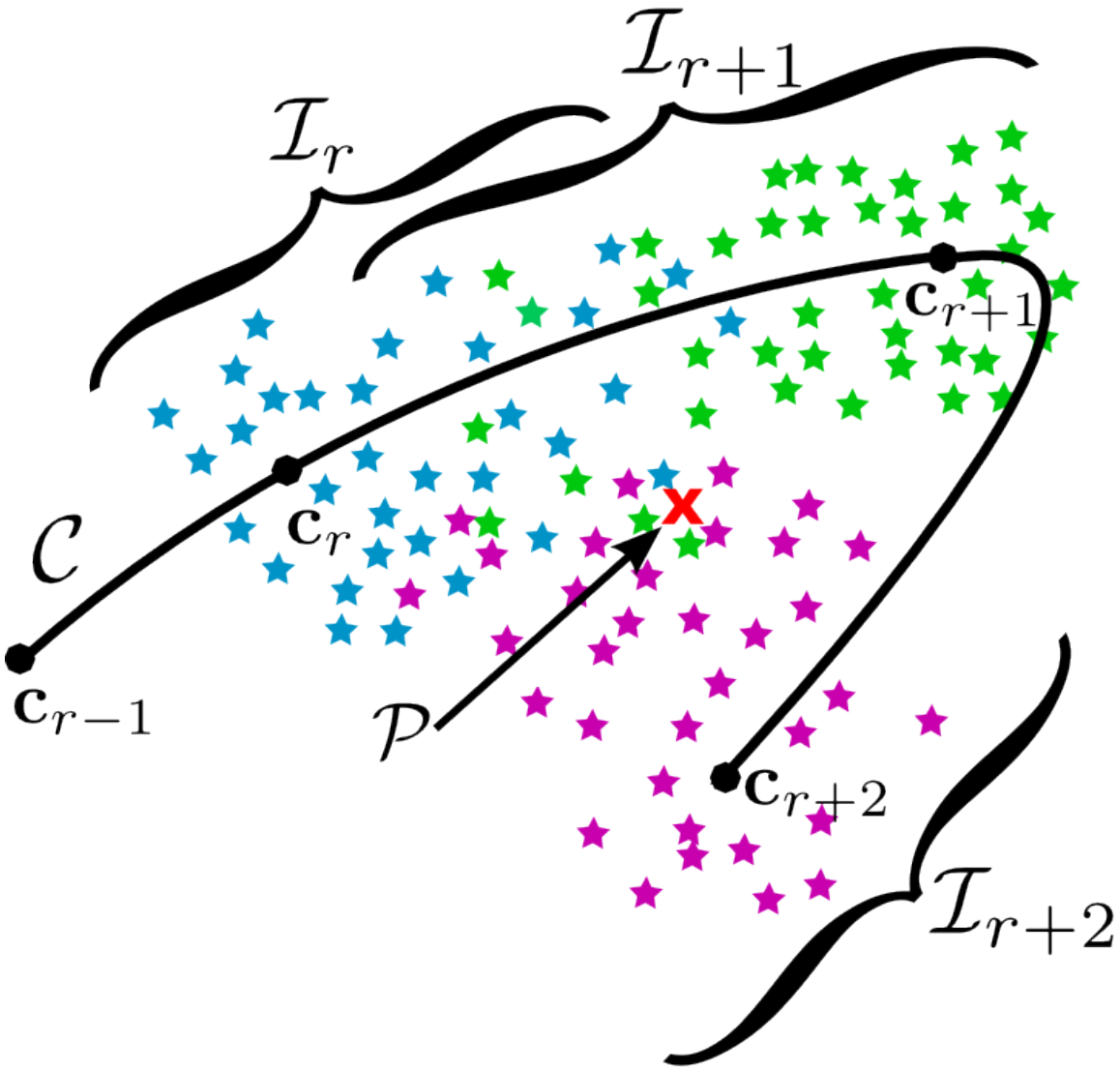

Figure 40: A graphical depiction of three illumination manifolds $\mathcal{I}_r$, $\mathcal{I}_{r+1}$, and $\mathcal{I}_{r+2}$ along with the projection of an input image $\mathcal{P}$ showing the difficulty of pose estimation encountered when the illumination manifolds intersect one another.

illumination manifolds $\mathcal{P}$ belongs to. Fortunately, the structure of the eigenspace manifold can be analyzed off-line to evaluate whether there is significant overlap between any of the a illumination manifolds.

To begin this analysis, the vector $\mathbf{p}_r^{r+1} = \mathcal{I}_r^T(\mathbf{c}_{r+1} - \mathbf{c}_r)$ is defined. This vector contains the orthogonal projections of $\mathcal{I}_r$ onto a linear approximation of the manifold $\mathcal{C}$ from $\mathbf{c}_r$ to $\mathbf{c}_{r+1}$. Similarly, the vector $\mathbf{p}_r^{r-1} = \mathcal{I}_r^T(\mathbf{c}_{r-1} - \mathbf{c}_r)$ is constructed to evaluate how close $\mathcal{I}_r$

is to $\mathcal{I}_{r-1}$. The maximum orthogonal projection from $\mathcal{I}_r$ to $\mathcal{I}_{r+1}$ and $\mathcal{I}_r$ to $\mathcal{I}_{r-1}$ is then computed by evaluating

$$p_r^{r+1} = \max_i(\mathbf{p}_r^{r+1}(i)) \quad \text{and} \quad p_r^{r-1} = \max_i(\mathbf{p}_r^{r-1}(i)), \tag{105}$$

respectively, where i is the i^{th} element of $\mathbf{p}_r^{r\pm1}$. A measure of how much the illumination manifolds overlap, i.e., how big the intersection of $\mathcal{I}_r$ is with $\mathcal{I}_{r-1}$ and $\mathcal{I}_{r+1}$, can now be evaluated by comparing the sum $p_r^{r+1} + p_{r+1}^r$ to $\Delta\mathcal{C}_r$, where $\Delta\mathcal{C}_r$ is the r^{th} element in $\Delta\mathcal{C}$, and $\Delta\mathcal{C} = [\|\mathbf{c}_2 - \mathbf{c}_1\|, \|\mathbf{c}_3 - \mathbf{c}_2\|, \cdots, \|\mathbf{c}_a - \mathbf{c}_{a-1}\|, \|\mathbf{c}_1 - \mathbf{c}_a\|]$ is the distance between each point in $\mathcal{C}$. In particular, if $\Delta\mathcal{C}_r \geq (p_r^{r+1} + p_{r+1}^r)$, then, $\mathcal{I}_r \cap \mathcal{I}_{r+1}$ is empty. An illustration of this is depicted in Fig. 41.

To evaluate this technique on real objects, the above analysis was used to evaluate the eigenspace structure for objects 1 and 13 from Fig. 17. Fig. 42 shows $\Delta\mathcal{C}$, $\mathbf{p}_r^{r+1}$, and $\mathbf{p}_r^{r-1}$ for both objects across all 90 poses. As can be seen from the figure, for object 1, $\Delta\mathcal{C}$ has good separation from both $\mathbf{p}_r^{r+1}$, and $\mathbf{p}_r^{r-1}$. As a result, the eigenspace structure (dimension and eigenimages) should allow for accurate pose estimation by simply using the centroid

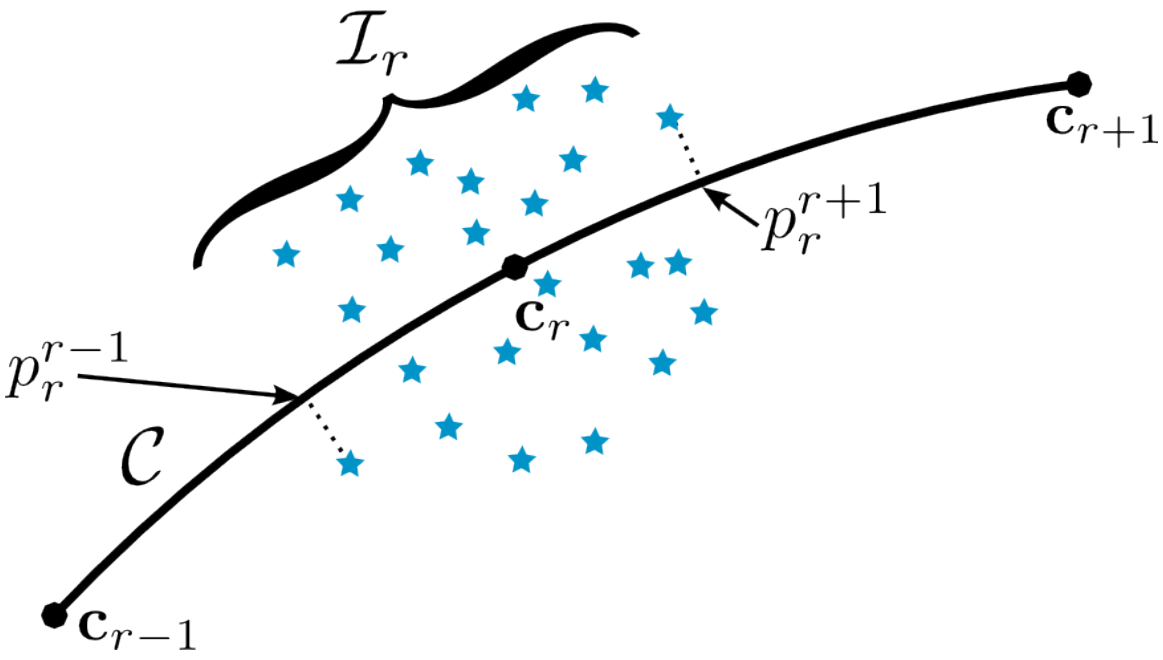

Figure 41: Illustration of three points on the manifold $\mathcal{C}$ showing the illumination cluster generated by $\mathcal{I}_r$. The projections p_r^{r+1} and p_r^{r-1} are also depicted.

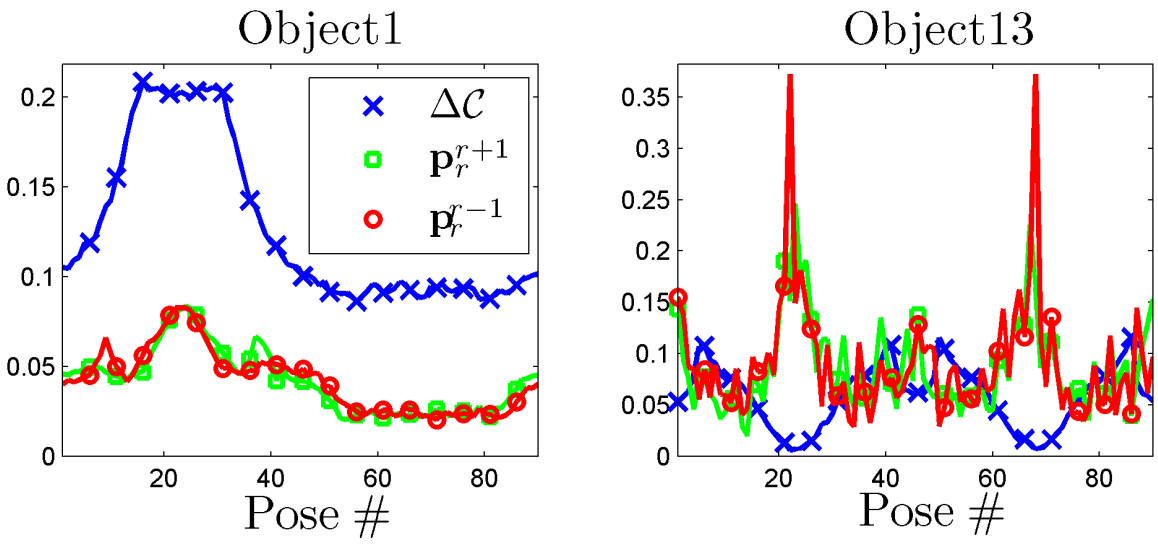

Figure 42: Plots of $\Delta\mathcal{C}$, $\mathbf{p}_r^{r+1}$, and $\mathbf{p}_r^{r-1}$ for objects 1 and 13.

manifold $\mathcal{C}$ for this object. An evaluation of object 13 however, reveals that accurate pose estimation for this object is unlikely for any of the 90 poses. This is because for this particular object, variation due to a change in pose is typically not larger than variation due to a change in illumination. Therefore, the intersections of the illumination manifolds $\mathcal{I}_r$ are significant across all 90 poses. Some of the reasons behind the multiple illumination manifold intersections for object 13 are a function of how the object is sampled. Recall that the camera is being moved along lines of constant co-latitude resulting in variation due to a change in pose being correlated on S^1. This particular object is is an LED that is being imaged around it's symmetric axis. As such, the only distinguishing feature in terms of a change in orientation comes from the leads of the LED. Unfortunately, the leads of the LED account for a very small percentage of total pixel area. Therefore, even large changes in orientation result in small changes in the image, i.e., the problem in its current is ill conditioned. On the other hand, if the object were imaged in it's stable orientation (laying flat), then much better results would be obtained.

8.4 Estimation Accuracy

To evaluate the accuracy of estimating the pose of objects using both the direct method (searching the entire eigenspace), and the proposed centroid manifold method, images of

the objects in Fig. 17 were captured from 90 random but known poses. The illumination conditions used for the test images was generated using the procedure outlined in Section 7.5. The algorithm described in Chapter 7 was then used to calculate $\tilde{\hat{U}}_k$ and $\mathcal{M}$ from a set of images of each object assuming a single illumination source and the sampling method depicted in Fig. 28. A test image was then projected onto it's corresponding eigenspace generating a new k-dimensional point $\mathcal{P}$. To determine which of the illumination manifolds $\mathcal{I}_r$ lies closest to the point $\mathcal{P}$ using traditional methods, a closest point search of the entire eigenspace manifold was performed. To determine which of the illumination manifolds $\mathcal{I}_r$ lies closest to the point $\mathcal{P}$ using the proposed method, the centroid manifold $\mathcal{C}$ was searched to determine which two points in $\mathcal{C}$ lie closest to the point $\mathcal{P}$. The entire illumination manifolds corresponding to these two points was then searched to determine which of the two manifolds lies closest to $\mathcal{P}$. This process was then repeated for each of the 270 test images (90 for each of the three illumination conditions). Note that using traditional search techniques results in an evaluation of n points, versus the $a + 2b$ points using the proposed technique. For the data used in this dissertation, this results in 4320 evaluations using traditional methods versus 186 evaluations using the proposed approach if exhaustive search techniques are employed.

The accuracy rate at which each search technique is capable of estimating the correct pose of the object for the 270 test images captured using a single illumination source (SS), two illumination sources (DS), and three illumination sources (TS) was then calculated, the results of which are outlined in Table 7. Note that the correct estimation rates for the proposed search technique are very comparable to the traditional search method. It is also interesting to note that for most objects, adding multiple illumination sources does not significantly reduce the accuracy of estimation rate (and in some instances it actually increases the rate of correct pose estimation). The data in Table 7 was also plotted in Fig. 43 to make comparing each technique more straightforward.

Because using a single source and searching the entire eigenspace manifold should produce the best possible result, i.e., the second column in Table 7, this provides a measure for determining the difficulty of accurate pose estimation for each object when variations

Table 7: Rate at which both search techniques are capable of estimating the correct pose of the object for 270 test images captured using a single illumination source (SS), two illumination sources (DS), and three illumination sources (TS).

Object	Direct [%]			Centroid [%]		
	SS	DS	TS	SS	DS	TS
1	95.56	92.22	90.00	95.56	92.22	90.00
2	77.78	65.56	62.22	62.22	56.67	55.56
3	88.89	90.00	86.67	86.67	87.78	86.67
4	96.67	92.22	92.22	96.67	92.22	92.22
5	91.11	91.11	87.78	91.11	91.11	87.78
6	94.44	93.33	93.33	94.44	93.33	93.33
7	86.67	76.67	78.89	86.67	75.56	77.78
8	97.78	95.56	91.11	97.78	95.56	91.11
9	96.67	96.67	91.11	96.67	96.67	91.11
10	95.56	95.56	96.67	95.56	95.56	96.67
11	94.44	94.44	92.22	94.44	94.44	92.22
12	90.00	93.33	90.00	90.00	93.33	90.00
13	27.78	22.22	24.44	18.89	18.89	34.44
14	91.11	88.89	84.44	91.11	88.89	84.44
15	95.56	93.33	90.00	95.56	93.33	90.00
16	68.89	65.56	72.22	62.22	63.33	68.89
17	95.56	87.78	92.22	95.56	87.78	91.11
18	90.00	80.00	70.00	87.78	77.78	68.89
19	96.67	90.00	88.89	88.89	84.44	84.44
20	84.44	75.56	71.11	82.22	73.33	65.56
Median	92.78	90.56	89.44	91.11	90.00	88.89
Min	27.78	22.22	24.44	18.89	18.89	34.44
Max	97.78	96.67	96.67	97.78	96.67	96.67

in illumination exist. Therefore, the proposed eigenspace search technique is compared to this data. The top plot in Fig. 43 shows rate of failure in pose identification for a single illumination source using the traditional search technique ordered from worst to best. The bottom plot in Fig. 43 shows the difference between the rate of success using the standard eigenspace search method and the proposed search method for a single illumination source (SS), two illumination sources (DS), and three illumination sources (TS). Note that pose estimation for some objects (13, 16, 2, and 20) is inherently difficult regardless of the search technique being employed. However, if one excludes such ill-conditioned cases, then the proposed search technique is very comparable to traditional methods in terms of being able to successfully identify the correct pose of the object.

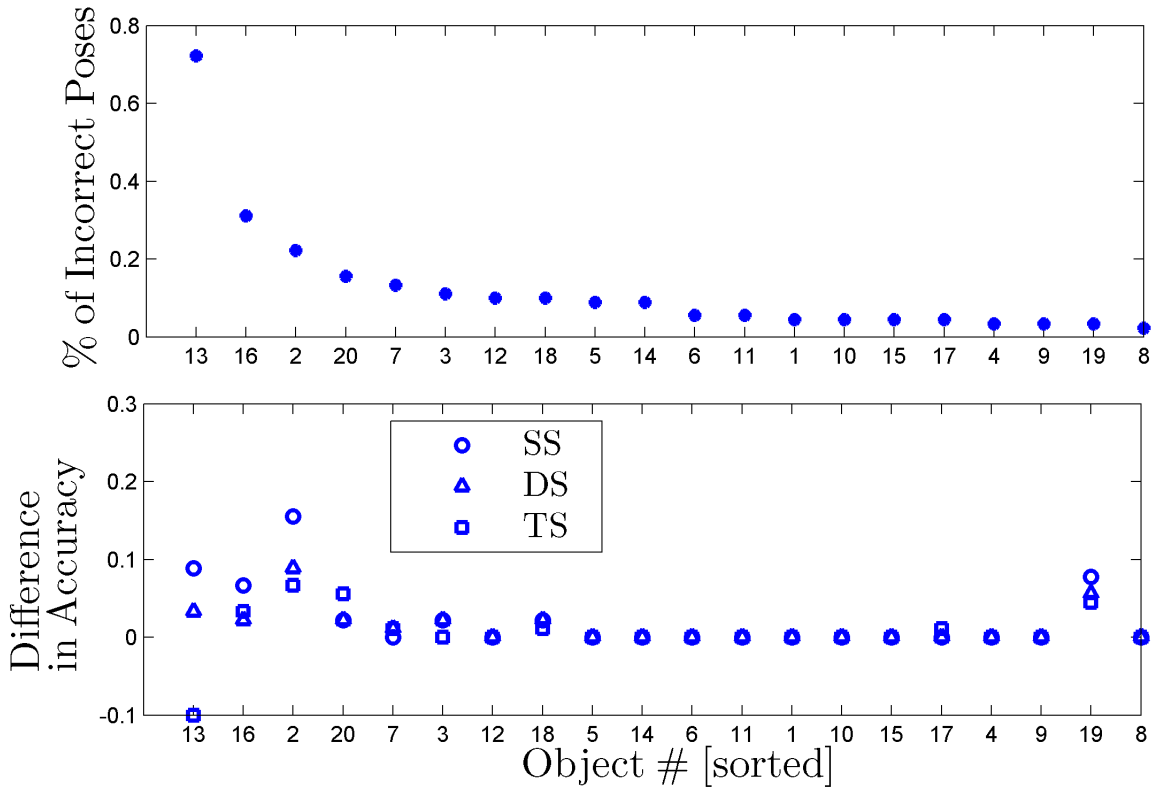

Figure 43: The rate of failure in pose estimation for a single illumination source using the traditional search technique, sorted from worst to best [top] and the difference between the rate of success using the standard eigenspace search method and the proposed search method for a single illumination source (SS), two illumination sources (DS), and three illumination sources (TS) [bottom].

CHAPTER IX

CONCLUSIONS

9.1 Summary

The first part of this dissertation presented a computationally efficient algorithm for estimating the subspace of images correlated in multiple dimensions for use in fully general 3-D pose estimation, assuming a single distant point light source. This required a method of sampling objects to produce a spherical data set by sampling S^2 and $SO(3)$ appropriately. It was illustrated that for data correlated on S^2, the spherical harmonic transform can be used to calculate the frequency information of this spherical data set. If the data is correlated on $SO(3)$, then spherical harmonics in conjunction with Wigner-D matrices can be used to capture the frequency information of this data set. A method to calculate the frequency information of hemispherically correlated data, which is useful in aerial pose estimation, was also presented. It was shown that computing the eigenspace decomposition on a subset of harmonic images in the transform domain can significantly reduce the computational burden. Experimental results were presented to validate both qualitatively and quantitatively the accuracy of the estimation, as well as the significant computational savings. Furthermore, the maximum energy recovery when using the spherical harmonic transform was quantified. It was shown that the errors associated with the proposed algorithm are minimal with respect to computational savings.

The second part of this dissertation dealt with the problem of pose estimation under variations in illumination from single and multiple illumination sources. It was illustrated that the dimensionality of a set of images of an object from a fixed viewpoint and multiple illumination conditions can be significantly reduced by means of the spherical harmonic

transform. Furthermore, it was shown that this reduced order subspace is capable of recovering a significant amount of information from images of objects where multiple illumination sources are present. In fact, it was shown that multiple illumination sources affect the eigenspace decomposition by increasing the probability that local specularities will be illuminated, thus decreasing the accuracy of the low-dimensional representation. An algorithm was then presented to compute the eigenspace decomposition of a set of images that contain variations in both pose as well as illumination. Experimental results were presented to validate the accuracy of the estimation, as well as the significant computational savings. Finally, an analysis of eigenspace manifolds was presented when variations due to a change in illumination and pose exist. The analysis showed that for most objects, variations due to a change in pose tend to be much larger than variations due to a change in illumination conditions. Based on this analysis, a technique was presented to perform illumination invariant pose estimation by using the centroid manifold to partition the eigenspace into smaller search regions. It was shown that using the proposed manifold search technique produced comparable results to traditional search methods. Furthermore, it was shown that although adding multiple illumination sources degrades the estimation rate, for most objects, the degradation is small.

9.2 Future Directions

One obvious future development would be to extend the algorithm developed in Chapter 7 to multiple dimensions. This should be a straight forward extension by using the algorithms developed in Chapters 5 and 6 to reduce the dimensionality of the data in the temporal dimension rather than using Chang's algorithm.

Another interesting future direction would be to evaluate images captured on the special Euclidean group $SE(3)$ and use a similar spectral decomposition to evaluate the spectral information. This would essentially produce a set of eigenimages that would have the interpretation of eigentrajectories. This would be extremely useful for target tracking problems.

Finally, the continuation of eigenspace manifold analysis, and manifold analysis in general, has the potential for solving several interesting open problems. First, using information

about the "true" image manifold may give significant insight into how its low-dimensional counter-part will perform. This information could also provide a more concrete method for choosing the dimension of the subspace, as well as the subspace itself, i.e., choosing the "principle" eigenimages may not be the optimal solution for every object/application. Manifold analysis could also provide significant insight into the problem of manifold learning for eigensystems. In particular, visualizing significant properties (both local and global) of the manifold embedded in high-dimensional space by means of a low-dimensional embedding.

REFERENCES

[1] K. Legaz. (2007) Kator Legaz: 3-D model database for Blender. [Online]. Available: http://www.katorlegaz.com/3d_models/index.php

[2] O. Selfridge and U. Neisser, "Pattern recognition by machine," *Scientific American*, vol. 203, pp. 60–68, Aug. 1960.

[3] L. G. Roberts, "Machine perception of three-dimensional solids," Ph.D. dissertation, Massachusetts Institute of Technology, 1963.

[4] U. Neisser and O. Selfridge, "Pattern recognition by machine," in *Computers and Thought*, 1963, pp. 237–250.

[5] R. T. Chin and C. R. Dyer, "Model-based recognition in robot vision," *Computing Surveys*, vol. 18, no. 1, pp. 67–108, Mar. 1986.

[6] C.-Y. Chang, "Eigenspace methods for correlated images," Ph.D. dissertation, Purdue University, 1999.

[7] K. Saitwal, "Fast eigenspace decomposition of correlated images using their spatial and temporal properties," Ph.D. dissertation, Colorado State University, 2006.

[8] H. C. Andrews and C. L. Patterson, "Singular value decomposition (SVD) image coding," *IEEE Trans. on Communications*, vol. 24, pp. 425–422, Apr. 1976.

[9] H. C. Andrews and C. L. Patterson, "Singular value decomposition and digital image processing," *IEEE Trans. ASSP*, vol. 24, no. 1, pp. 26–53, Feb. 1976.

[10] Y. S. Shim and Z. H. Cho, "SVD pseudoinversion image reconstruction," *IEEE Trans. ASSP*, vol. 24, no. 4, pp. 904–909, Aug. 1981.

[11] K. Fukunaga, *Introduction to Statistical Pattern Recognition.* London, U.K.: Academic, 1990.

[12] J. J. Gerbrands, "On the relationships between SVD, KLT and PCA," *Pattern Recognition*, vol. 14, no. 1-6, pp. 375–381, Dec. 1981.

[13] M. Kirby and L. Sirovich, "Application of the Karhunen-Loeve procedure for the characterization of human faces," *IEEE Trans. PAMI*, vol. 12, no. 1, pp. 103–108, Jan. 1990.

[14] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cogn. Neurosci.*, vol. 3, no. 1, pp. 71–86, Mar. 1991.

[15] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. PAMI*, vol. 19, no. 7, pp. 711–720, July 1997.

[16] R. Brunelli and T. Poggio, "Face recognition: Features versus templates," *IEEE Trans. PAMI*, vol. 15, no. 10, pp. 1042–1052, Oct. 1993.

[17] A. Pentland, B. Moghaddam, and T. Starner, "View-based and modular eigenspaces for face recognition," in *IEEE Conf. Comp. Vis. and Patt. Rec.*, Seattle, WA, June 1994, pp. 84–91.

[18] M. H. Yang, D. J. Kriegman, and N. Ahuja, "Detecting faces in images: A survey," *IEEE Trans. PAMI*, vol. 24, no. 1, pp. 34–58, Jan. 2002.

[19] H. Murase and R. Sakai, "Moving object recognition in eigenspace representation: Gait analysis and lip reading," *Patt. Rec. Lett.*, vol. 17, no. 2, pp. 155–162, Feb. 1996.

[20] G. Chiou and J. N. Hwang, "Lipreading from color video," *IEEE Trans. Image Proc.*, vol. 6, no. 8, pp. 1192–1195, Aug. 1997.

[21] H. Murase and S. K. Nayar, "Illumination planning for object recognition using parametric eigenspaces," *IEEE Trans. PAMI*, vol. 16, no. 12, pp. 1219–1227, Dec. 1994.

[22] C. Y. Huang, O. I. Camps, and T. Kanungo, "Object recognition using appearance-based parts and relations," in *IEEE Conf. Comp. Vis. and Patt. Rec.*, San Juan, PR, June 1997, pp. 877–883.

[23] R. J. Campbell and P. J. Flynn, "Eigenshapes for 3-D object recognition in range data," in *IEEE Conf. Comp. Vis. and Patt. Rec.*, Fort Collins, CO, June 1999, pp. 505–510.

[24] H. Murase and S. K. Nayar, "Visual learning and recognition of 3-D objects from appearance," *Int. J. Comp. Vis.*, vol. 14, no. 1, pp. 5–24, Jan. 1995.

[25] S. K. Nayar, S. A. Nene, and H. Murase, "Subspace methods for robot vision," *IEEE Trans. Robot. Automat.*, vol. 12, no. 5, pp. 750–758, Oct. 1996.

[26] C. Y. Chang, A. A. Maciejewski, and V. Balakrishnan, "Fast eigenspace decomposition of correlated images," *IEEE Trans. Image Proc.*, vol. 9, no. 11, pp. 1937–1949, Nov. 2000.

[27] R. C. Hoover, A. A. Maciejewski, and R. G. Roberts, "Aerial pose detection of 3-D objects using hemispherical harmonics," in *IEEE SSIAI*, Santa Fe, NM, Mar. 2008, pp. 157–160.

[28] R. C. Hoover, A. A. Maciejewski, and R. G. Roberts, "Pose detection of 3-D objects using S^2-correlated images and discrete spherical harmonic transforms," in *IEEE Int. Conf. Robot. Automat.*, Pasadena, CA, May. 2008, pp. 993–998.

[29] R. C. Hoover, A. A. Maciejewski, and R. G. Roberts, "Pose detection of 3-D objects using images sampled on $SO(3)$, spherical harmonics, and Wigner-D matrices," in *IEEE Conf. on Automat. Sci. and Engr.*, Washington DC, Aug 2008, pp. 47–52.

[30] R. C. Hoover, A. A. Maciejewski, R. G. Roberts, and R. P. Hoppal, "An illustration of eigenspace decomposition for illumination invariant pose estimation," in *IEEE Int. Conf. on Sys., Man, and Cyb.*, San Antonio, TX, accepted to appear Oct. 2009. [Online]. Available: http://www.engr.colostate.edu/~hooverr/Drafts/SMC_1.pdf

[31] R. C. Hoover, A. A. Maciejewski, and R. G. Roberts, "Eigendecomposition of Images Correlated on S^1, S^2, and $SO(3)$ Using Spectral Theory," accepted to appear in *IEEE Trans. Image Proc.*, 2009.

[32] E. N. Malamasa, E. G. Petrakisa, M. Zervakisa, L. Petit, and J.-D. Legat, "A survey on industrial vision systems, applications and tools," *Image and Vision Comp.*, vol. 21, no. 2, pp. 171–188, 2003.

[33] H. Murakami and V. Kumar, "Efficient calculation of primary images from a set of images," *IEEE Trans. PAMI*, vol. 4, no. 5, pp. 511–515, Sept. 1982.

[34] I. J. Good, "Some applications of the singular value decomposition of a matrix," *Technometrics*, vol. 11, pp. 823–831, Nov. 1969.

[35] S. Shlien, "A method for computing the partial singular value decomposition," *IEEE Trans. PAMI*, vol. 4, no. 6, pp. 671–676, Nov. 1982.

[36] R. Haimi-Cohen and A. Cohen, "Gradient-type algorithms for partial singular value decomposition," *IEEE Trans. PAMI*, vol. 9, no. 1, pp. 137–142, Jan. 1987.

[37] X. Yang, T. K. Sarkar, and E. Arvas, "A survey of conjugate gradient algorithms for solution of extreme eigen-problems for a symmetric matrix," *IEEE Trans. ASSP*, vol. 37, no. 10, pp. 1550–1556, Oct. 1989.

[38] C. R. Vogel and J. G. Wade, "Iterative SVD-based methods for ill-posed problems," *SIAM J. Sci. Comput.*, vol. 15, no. 3, pp. 736–754, May 1994.

[39] S. Chandrasekaran, B. Manjunath, Y. Wang, J. Winkeler, and H. Zhang, "An eigenspace update algorithm for image analysis," *CVGIP: Graphic Models and Image Proc.*, vol. 59, no. 5, pp. 321–332, Sept. 1997.

[40] K. Saitwal, A. A. Maciejewski, and R. G. Roberts, "Computationally efficient eigenspace decomposition of correlated images characterized by three parameters," accepted to appear in *Patt. Anal. and Apps.*, 2009.

[41] K. Saitwal, A. A. Maciejewski, R. G. Roberts, and B. Draper, "Using the low-resolution properties of correlated images to improve the computational efficiency of eigenspace decomposition," *IEEE Trans. Image Proc.*, vol. 15, no. 8, pp. 2376–2387, Aug. 2006.

[42] G. H. Golub and C. F. V. Loan, *Matrix Computations*. Baltimore, MD: Johns Hopkins, 1996.

[43] E. Isaacson and H. B. Keller, *Analysis of Numerical Methods*. New York City, NY: Wiley, 1966.

[44] G. W. Stewart, *Introduction to Matrix Computation*. New York City, NY: Academic, 1973.

[45] H. Murase and M. Lindenbaum, "Partial eigenvalue decomposition of large images using the spatial temporal adaptive method," *IEEE Trans. Image Proc.*, vol. 4, no. 5, pp. 620–629, May. 1995.

[46] R. C. Hoover, A. A. Maciejewski, and R. G. Roberts, "An analysis of sphere tessellations for pose estimation of 3-D objects using spherically correlated images," in *IEEE SSIAI*, Santa Fe, NM, Mar. 2008, pp. 41–44.

[47] P. N. Swarztrauber, "On the spectral approximation of discrete scalar and vector functions on the sphere," *SIAM J. Numer Anal.*, vol. 16, no. 6, pp. 934–949, Dec. 1979.

[48] P. N. Swarztrauber and W. F. Spotz, "Generalized discrete spherical harmonic transforms," *J. of Comp. Phys.*, vol. 159, no. 2, pp. 213–230, Apr. 2000.

[49] P. N. Swarztrauber, "On computing the points and weights for Gauss-Legendre quadrature," *SIAM J. on Sci. Comp.*, vol. 24, no. 3, pp. 945–954, 2002.

[50] J. R. Driscoll and D. M. Healy Jr., "Computing Fourier transforms and convolutions on the 2-sphere," *Adv. App. Math*, vol. 15, no. 2, pp. 202–250, June 1994.

[51] D. M. Healy Jr., D. Rockmore, P. Kostelec, and S. Moore, "FFTs for the 2-sphere-improvements and variations," *J. of Fourier Anal. and App.*, vol. 9, no. 4, pp. 341–385, July 2003.

[52] K. M. Górski, E. Hivon, A. L. Banday, B. D. Wandelt, F. K. Hansen, M. Reinecke, and M. Bartelmann, "HEALPix: A framework for high-resolution discretization and

fast analysis of data distributed on the sphere," *The Astrophysical Journal*, vol. 622, pp. 759–771, Apr. 2005.

[53] Y. Wiaux, L. Jacques, P. Vielva, and P. Vandergheynst, "Fast directional correlation on the sphere with steerable filters," *The Astrophysical Journal*, vol. 652, p. 820, Sept. 2006.

[54] P. N. Swarztrauber, "The vector harmonic transform method for solving partial differential equations in spherical geometry," *Monthly Weather Review*, vol. 121, pp. 3415–3437, May 1993.

[55] M. A. Wieczorek and F. J. Simons, "Localized spectral analysis on the sphere," *Geoph. J. Int*, vol. 162, no. 3, pp. 655–675, Sept. 2005.

[56] F. J. Simons, F. A. Dahlen, and M. A. Wieczorek, "Spatiospectral localization on a sphere," *SIAM Review*, vol. 48, no. 3, pp. 504–536, 2006.

[57] D. A. Varshalovich, A. N. Moskalev, and V. K. Khersonskii, *Quantum Theory of Angular Momentum.* Hackensack, NJ: World Scientific, 1988.

[58] G. B. Arfken, *Mathematical Methods for Physicists.* New York City, NY: Academic Press, 1970.

[59] G. S. Chirikjian and A. B. Kyatkin, *Engineering Applications of Noncommutative Harmonic Analysis: With Emphasis on Rotation and Motion Groups.* Boca Raton, FL: CRC Press, 2001.

[60] R. Ramamoorthi, "Analytic PCA construction for theoretical analysis of lighting variability in images of a Lambertian object," *IEEE Trans. PAMI*, vol. 24, no. 10, pp. 1322–1333, Oct. 2002.

[61] L. Qing, S. Shan, and W. Gao, "Eigen-harmonic faces: Face recognition under generic lighting," in *IEEE Int. Conf. Auto. Face Gesture Rec.*, Seoul, Korea, May 2004, pp. 296–301.

[62] S. Romdhani, J. Ho, T. Vetter, , and D. J. Kriegman, "Face recognition using 3-D models: Pose and illumination," *Proc. of the IEEE*, vol. 294, no. 11, pp. 1977–1999, Nov. 2006.

[63] L. Zhang and D. Samaras, "Face recognition from a single training image under arbitrary unknown lighting using spherical harmonics," *IEEE Trans. PAMI*, vol. 28, no. 3, pp. 351–363, Mar. 2006.

[64] B. Bustos, D. A. Keim, D. Saupe, T. Schreck, and D. V. Vranic, "Feature-based similarity search in 3-D object databases," *ACM Comp. Surveys*, vol. 37, no. 4, pp. 345–387, Dec. 2005.

[65] D. Saupe and D. V. Vranic, "3-D model retrieval with spherical harmonics and moments," in *Deutsche Arbeitsgemeinschaft für Mustererkennung (DAGM)-Symp. on Patt. Rec.*, Munich, Germany, Sept. 2001, pp. 392–397.

[66] D. V. Vranic, "An improvement of rotation invariant 3d-shape based on functions on concentric spheres," in *IEEE Conf. Image Proc.*, Barcelona, Spain, Sept. 2003, pp. 757–760.

[67] A. Makadia and K. Daniilidis, "Rotation recovery from spherical images without correspondences," *IEEE Trans. PAMI*, vol. 28, no. 7, pp. 1170–1175, July 2006.

[68] M. J. Mohlenkamp, "A fast transform for spherical harmonics," Ph.D. dissertation, Yale University, 1997.

[69] E. Stein and G. Weiss, *Introduction to Fourier Analysis on Euclidean Spaces.* Princton, NJ: Princton University Press, 1971.

[70] M. J. Mohlenkamp, "A Users Guide to Spherical Harmonics," http://www.math.ohiou.edu/~mjm/research/uguide.pdf, 2008.

[71] N. Sneeuw, "Global spherical harmonic analysis by least squares and numerical quadrature methods in historical perspective," http://www.geomatics.ucalgary.ca/~sneeuw/publ/neumann.ps, 2007.

[72] P. Gautron, J. Křivánek, S.Pattanaik, and K. Bouatouch, "A novel hemispherical basis for accurate and efficient rendering," in *Eurographics Symposium on Rendering*, 2004, pp. 321–330.

[73] P. J. Kostelec and D. N. Rockmore, "FFTs on the Rotation Group," Santa Fe Institute Working Papers Series Paper #03-11-060, 2003.

[74] E. P. Wigner, *Group Theory and its Application to the Quantum Mechanics of Atomic Spectra*. New York City, NY: Academic Press, 1959.

[75] M. Blanco, M. Florez, and M. Bermejo, "Evaluation of the rotation matrices in the basis of real spherical harmonics," *Journal of Molecular Structure: THEOCHEM*, vol. 419, pp. 19–27, Dec. 1997.

[76] J. Ivanic and K. Ruedenberg, "Rotation matrices for real spherical harmonics. Direct determination by recursion," *J. Phys. Chem*, vol. 100, no. 15, pp. 6342–6347, Apr. 1996.

[77] T. Risbo, "Fourier transform summation of Legendre series and D-functions," *J. of Geodesy*, vol. 70, no. 7, pp. 383–396, 1996.

[78] A. Shashua, "On photometric issues in 3D visual recognition from a single 2D image," *Int. J. Comp. Vis.*, vol. 21, pp. 99–122, Jan. 1997.

[79] P. Hallinan, "A low-dimensional representation of human faces for arbitrary lighting conditions," in *IEEE Conf. on Comp. Vis. and Patt. Recog.*, Seattle, WA, June 1994, pp. 995–999.

[80] R. Epstein, P. Hallinan, and A. Yuille, "5 $\pm$ 2 eigenimages suffice: An empirical investigation of low-dimensional lighting models," in *IEEE Workshop on Physics-Based Vision*, Cambridge, MA, June 1995, pp. 108–116.

[81] P. N. Belhumeur and D. J. Kriegman, "What is the set of images of an object under all possible illumination conditions?" *Int. J. of Comp. Vis.*, vol. 28, no. 3, pp. 245–260, July 1998.

[82] D. J. Kriegman, P. N. Belhumeur, and A. S. Georghiades, "Representations for recognition under variable illumination," in *Shape, Contour and Grouping in Computer Vision*, 1999, pp. 95–131.

[83] R. Basri and D. Jacobs, "Lambertian reflectance and linear subspaces," *IEEE Trans. PAMI*, vol. 25, pp. 383–390, Feb. 2003.

[84] S. K. Nayar and H. Murase, "Dimensionality of illumination in appearance matching," in *IEEE Int. Conf. Robot. Automat.*, Minneapolis, MN, Apr. 1996, pp. 1326–1332.

[85] A. Georghiades, D. Kriegman, and P. Belhumeur, "Illumination cones for recognition under variable lighting: Faces," in *IEEE Conf. on Comp. Vis. and Patt. Recog.*, Santa Barbara, CA, June 1998, pp. 52–59.

[86] K. Lee, J. Ho, and D. Kriegman, "Acquiring linear subspaces for face recognition under variable lighting," *IEEE Trans. PAMI*, vol. 27, pp. 684–698, May 2005.

[87] (2009) Blender: Open source 3-D modeling application. [Online]. Available: http://www.blender.org/

[88] K. Fukunage and P. M. Narendra, "A branch and bound algorithm for computing k-nearest neighbors," *IEEE Trans. Comput.*, vol. 24, no. 7, pp. 750–753, July 1975.

[89] J. L. Bentley, "Multidimensional binary search trees in database applications," *IEEE Trans. Softw. Eng.*, vol. 5, no. 4, pp. 333–340, July 1979.

[90] A. Guttman, "R-trees: A dynamic index structure for spatial searching," in *SIGMOD, Proceedings of Annual Meeting*, Boston, MA, June 1984, pp. 47–57.

[91] F. Aurenhammer, "Voronoi diagrams: A survey of a fundamental geometric data structure," *ACM Computing Surveys*, vol. 23, no. 3, pp. 345–405, Sept. 1991.

[92] A. Califano and R. Mohan, "Multidimensional indexing for recognizing visual shapes," *IEEE Trans. PAMI*, vol. 16, no. 4, pp. 373–392, Apr. 1994.

[93] K. L. Clarkson, "A randomized algorithm for closest-point queries," *SIAM J. Comput.*, vol. 17, no. 4, pp. 830–847, Aug. 1988.

[94] S. Nene and S. Nayar, "A simple algorithm for nearest neighbor search in high dimensions," *IEEE Trans. PAMI*, vol. 19, no. 9, pp. 989–1003, Sept. 1997.

[95] R. C. Hoover, A. A. Maciejewski, and R. G. Roberts, "Designing eigenspace manifolds: With application to object identification and pose estimation," in *IEEE Int. Conf. on Sys., Man, and Cyb.*, San Antonio, TX, accepted to appear Oct. 2009. [Online]. Available: http://www.engr.colostate.edu/~hooverr/Drafts/SMC_2.pdf

[96] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, Dec. 2000.

[97] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, Dec. 2000.

[98] L. K. Saul, S. T. Roweis, and Y. Singer, "Think globally, fit locally: Unsupervised learning of low dimensional manifolds," *Journal of Machine Learning Research*, vol. 4, pp. 119–155, Dec. 2003.