DISSERTATION

GEOMETRIC METHODS ON SPECIAL MANIFOLDS FOR VISUAL RECOGNITION

Submitted by

Yui Man Lui

Department of Computer Science

In partial fulfillment of the requirements

for the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Spring 2010

COLORADO STATE UNIVERSITY

March 31, 2010

WE HEREBY RECOMMEND THAT THE DISSERTATION PREPARED UNDER OUR SUPERVISION BY YUI MAN LUI ENTITLED "GEOMETRIC METHODS ON SPECIAL MANIFOLDS FOR VISUAL RECOGNITION" BE ACCEPTED AS FULFILLING IN PART REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY.

Committee on Graduate Work

Bruce Draper

Michael Kirby

L. Darrell Whitley

Advisor: J. Ross Beveridge

Department Head: L. Darrell Whitley

ABSTRACT OF DISSERTATION

GEOMETRIC METHODS ON SPECIAL MANIFOLDS FOR VISUAL RECOGNITION

Many computer vision methods assume that the underlying geometry of images is Euclidean. This assumption is generally not valid. Therefore, this dissertation introduces new nonlinear geometric frameworks based upon special manifolds, namely Graßmann and Stiefel manifolds, for visual recognition. The motivation for this thesis is driven by the intrinsic geometry of visual data in which the visual data can be either a still image or video. Visual data are represented as points in appropriately chosen parameter spaces. The idiosyncratic aspects of the data in these spaces are then exploited for pattern classification. Three major research results are presented in this dissertation: face recognition for illumination spaces on Stiefel manifolds, face recognition on Graßmann registration manifolds, and action classification on product manifolds.

Previous work has shown that illumination cones are idiosyncratic for face recognition in illumination spaces. However, it has not been addressed how a single image relates to an illumination cone. In this dissertation, a Bayesian model is employed to relight a single image to a set of illuminated variants. The subspace formed by these illuminated variants is characterized on a Stiefel manifold. A new distance measure called Canonical Stiefel Quotient (CSQ) is introduced. CSQ performs two projections on a tangent space of a Stiefel manifold and uses the quotient for classification. The proposed method demonstrates that illumination cones can be synthesized by relighting a single image to a set of images, and the synthesized illumination cones are discriminative for face recognition. Experiments on the CMU-PIE and YaleB data sets reveal that CSQ not only achieves high recognition accuracies for generic faces but also is robust to the choice of training sets.

Subspaces can be realized as points on Graßmann manifolds. Motivated by image perturbation and the geometry of Graßmann manifolds, we present a method called Graßmann Registration Manifolds (GRM) for face recognition. First, a tangent space is formed by a set of affine perturbed images where the tangent space admits a vector space structure. Second, the tangent spaces are embedded on a Graßmann manifold and chordal distance is used to compare subspaces. Experiments on the FERET database suggest that the proposed method yields excellent results using both holistic and local features. Specifically, on the FERET $Dup2$ data set, which is generally considered the most difficult data set on FERET, the proposed method achieves the highest rank one identification rate among all non-trained methods currently in the literature.

Human actions compose a series of movements and can be described by a sequence of video frames. Since videos are multidimensional data, data tensors are the natural choice for data representation. In this dissertation, a data tensor is expressed as a point on a product manifold and classification is performed on this product space. First, we factorize a data tensor using a modified High Order Singular Value Decomposition (HOSVD) and recognize each factorized space as a Graßmann manifold. Consequently, a data tensor is mapped to a point on a product manifold and the geodesic distance on the product manifold is computed for tensor classification. The proposed method is geometrically sound and the metric is naturally inherited from the factor manifolds. Experiments on the Cambridge-Gesture and KTH human action data sets show that the proposed method outperforms the current state-of-the-art.

The use of special manifolds for visual recognition has just emerged. This dissertation shows that the underlying geometry of space is an important feature for pattern recognition. The proposed geometric frameworks are particularly suitable for high dimensional data, and will lead to many possible future work.

Yui Man Lui
Department of Computer Science
Colorado State University
Fort Collins, CO 80523
Spring 2010

ACKNOWLEDGEMENTS

Five year efforts are coming to a conclusion. I would like to take this opportunity to express my gratitude to the people who made this dissertation possible.

My deepest gratitude goes to my advisor, Dr. Ross Beveridge. I have been very fortunate to have Ross as my advisor. He gave me the freedom to explore my own research interests while guided me through rigorous assessments. His constant encouragement, patience, and support helped me overcome many obstacles and made an ordinary student like me to accomplish extraordinary works.

My committee members, Dr. Bruce Draper, Dr. Michael Kirby, and Dr. Darrell Whitley, have provided me many guidance and constructive comments. They helped me keep my research in high standards. I am indebted for their advices.

Special thanks go to Mrs. Elaine Regelson, who helped me settle down in academia when I returned from industry. Her kindhearted assistance made me feel like home in the computer science department.

My colleagues, Mr. David Bolme and Mr. Steve O'Hara, have accompanied me for lunch during my Ph.D. tenure. I have benefited from their expertise throughout our brain-storming discussion, and that has broadened my academic horizons.

My acknowledgment is also due to Mr. William Wong and Mrs. Elaine Fung. Their kindliness and care kept me sane through these years. The annual ski trip is definitely the much needed sustenance. I greatly value their friendship.

Lastly, I like to express my sincere appreciation to my mother. Without her love, none of this would have been possible. Thanks also go to my sister and my brother who have been taking care of the family over these years.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

This dissertation studies the intrinsic geometry of image space and proposes novel geometric frameworks based upon special manifolds, namely Stiefel and Graßmann manifolds, for visual recognition. The underlying geometry of visual data induces the development for the proposed frameworks. We realize an image-set as a point in a suitable parameter space and exploit the idiosyncratic nature in this parameter space for pattern recognition. The utilities of the proposed frameworks are demonstrated through the applications of face recognition and action classification.

This chapter describes the motivation for our studies, the idea of data abstraction, and the intuition of manifolds, particularly special manifolds. In addition, the main contributions of this research are highlighted and the subsequent chapters are introduced.

1.1 Motivation

Pattern recognition is one of the key attributes of intelligent behavior. Both humans and animals rely heavily on this ability for survival, for example, searching for food, avoiding hazard, and finding a mate for reproduction. All of these activities involve pattern recognition. However, patterns exhibit a huge amount of variability in nature. This variability could be caused by deformation, external sources, and / or aging. Hence, pattern recognition signifies a high level of adaptation and intelligent behavior.

Much of what is studied in the field of computer vision can be reduced to visual pattern recognition. A computer is viewed as an intelligent machine when it performs some tasks like

a human does. In computer vision, these tasks include object detection, face recognition, action classification, etc. Over the years, many efforts have been dedicated to mimicing the human visual system. These efforts build upon the foundation of mathematics which has played an important role in advancing the field of computer vision.

Among the mathematics used in computer vision, the geometry of space may be the most fundamental. Geometry arises naturally in computing the distance or similarity between patterns. Prior to pattern recognition / discrimination, one needs to represent a pattern in some parameter space. Previous methods often make an implicit or explicit assumption that the underlying geometry of patterns is Euclidean. For example, distance is often measured by a L_2 norm. This means that visual data have been characterized in a vector space. This characterization gives rise to an important question. Are images sampled from a linear space or a nonlinear space?

This question leads to the development of our geometric frameworks. Before we discuss the proposed paradigm, we discuss the principle of data abstraction and image representation in a topological space.

1.2 Data Abstraction

Data abstraction is the fundamental concept of our geometric frameworks. It is known that a point in a three dimensional space can be represented by a vector with three elements, $[x_1, x_2, x_3]^T$ in a Euclidean space depicted in Figure 1.1. Similarly, an image can be vectorized and represented by a vector with n elements, one element per pixel. As such, an image can be identified as a point in some topological space that we call image space. Figure 1.2 illustrates the association between an image and a point in the image space.

The space where vectors reside is called a vector space. It is a mathematically well-defined space which is closed under addition and scalar multiplication. Since the field operations are linear, the vector space is a linear space where the law of superposition applies. Any element in the vector space can be written as a product sum of other elements in the space, hence the space is open but the elements in a vector space are closed under linear operations and can be represented by a finite number of bases.

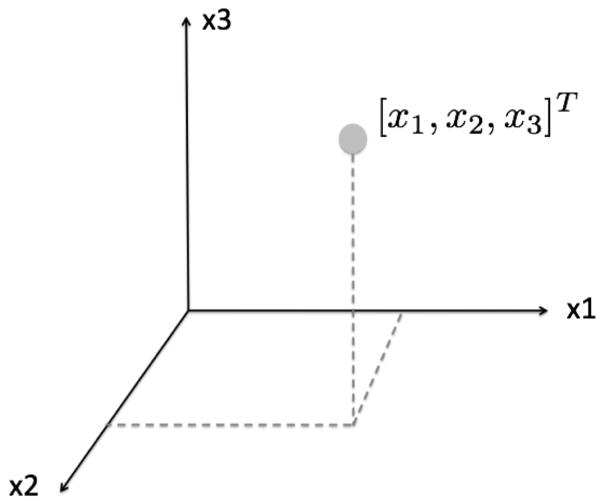


Figure 1.1: A point in a three dimensional space

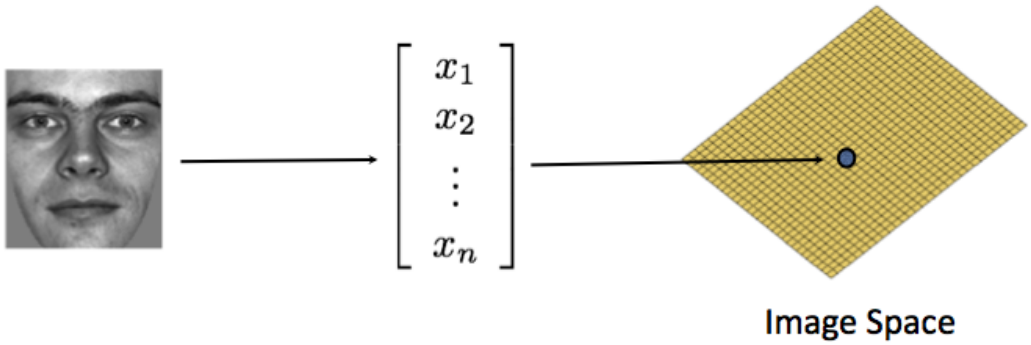


Figure 1.2: Viewing an image as a point in the image space

While the law of superposition holds for illumination spaces under a fixed pose and convex object assumptions [9], it does not apply to the nonlinear sources of image variability such as registration or deformation [96]. We argue that image space is generally not Euclidean because the law of superposition does not hold in this space. For example, there is no linear combinations to form a simple image translation over the field $\mathbb{R}$. Hence, the image space is generally nonlinear. Therefore, classification algorithms carried out in a Euclidean space may not have inherent the nonlinear geometric structure, as a result, the classification accuracy may be poor. To properly represent images in a parameter space, we turn our attention to nonlinear surfaces called manifolds.

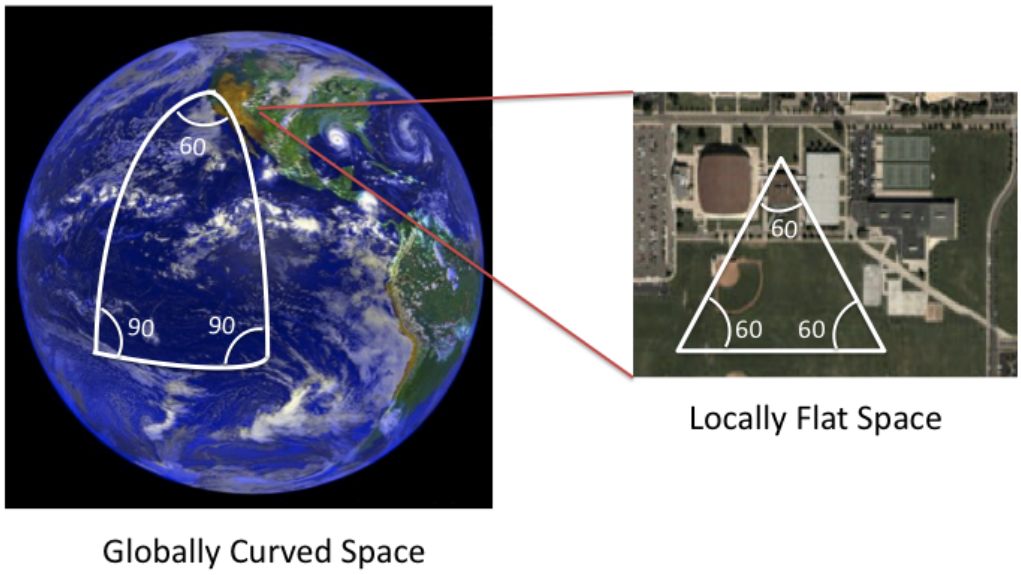


Figure 1.3: A two dimensional manifold embedded in $\mathbb{R}^3$

1.3 What is a manifold

Many years ago, people thought that we were living on a flat surface. Now, everyone knows that the Earth is a spherical object . Our Earth is a simple example of a compact manifold which is a globally curved but locally flat surface. Figure 1.3 illustrates that the surface of the Earth is a two dimensional manifold since it can be represented by a collection of two dimensional flat surfaces embedded in a three dimensional Euclidean space. Because the manifold is only regionally flat, the Euclidean geometry is only applicable locally.

It is important that the geometry of space is properly measured such that distances and sizes are calculated accordingly. Let us take our Earth as another example. Because of the curvature of the Earth, it takes less time to fly from New York to Hong Kong over the north pole than over San Francisco. Hence, any measure failing to take the spherical nature of the Earth is going to make mistakes. Likewise, any approach to pattern recognition neglecting the nonlinear geometry of the manifold will make mistakes as well.

From the differential geometry perspective, manifolds are sets with coordinate systems and can be viewed as smooth, curved surfaces embedded in high dimensional Euclidean spaces.

Furthermore, manifolds are locally similar (homeomorphic) to an open set in a Euclidean space. The distance is geodesic which is the shortest distance between two points and the geodesic path stays on the constraint surface (manifold), consequently the curvature of the space is taken into account.

Special manifolds, Stiefel and Graßmann manifolds, are differentiable manifolds with well-defined mathematical properties, and often represent natural data in computer vision, particularly image-sets. The Stiefel and Graßmann manifolds are sometimes referred to as embedded sub-manifolds of $\mathbb{R}^{n\times p}$ and quotient manifolds of $\mathbb{R}^{n\times p}$, respectively [1], where n is the embedded dimension and p is the manifold dimension. The special manifolds can be viewed as quotient spaces[1] of a special orthogonal group under different equivalence relations. This representation allows one to characterize certain elements from a set using an equivalence relation, and the set of equivalence classes[2] forms a quotient space. The equivalence classes on special manifolds induce some nice mathematical properties and make geodesic distance computable. Our geometric frameworks rest upon the properties of special manifolds.

1.4 Challenges

Special manifolds are a set of matrices in $\mathbb{R}^{n\times p}$ with orthogonality constraints. The visual applications of matrix manifolds are naturally related to image-sets where n is the number of pixels and p is the number images. The image-set based classification on Graßmann manifolds has been proposed by [18, 67, 11]. Cheng et al. [18] and Beveridge et al. [11] performed image-set classification in illumination spaces. A collection of face illumination variants taken under a fixed pose is viewed as an element on Graßmann manifolds. Lui et al. [67] employed geodesic distance to match image-sets in conjunction with cohort normalization on a Face Recognition Grand Challenge (FRGC) dataset. Although excellent results have been achieved, these meth-

[1] Quotient spaces will be introduced in chapter 3.

[2] Equivalence classes will be introduced in chapter 3.

ods are restricted to multiple images and they may not be available in some applications. The challenges therefore arise when we relax the image-set constraint to a single image or extend to a video containing time information.

This dissertation emphasizes the geometry of visual data for the purpose of classification. We relax the image-set constraint to a single image. New geometric frameworks called Canonical Stiefel Quotient (CSQ) and Graßmann Registration Manifolds (GRM) are presented for face recognition on special manifolds using a single image per set. To achieve this objective, image perturbation is applied to render a single image set to a multi-image set. In CSQ, a statistical model is employed to relight a single image to a set of illumination variants. In GRM, a registration manifold is sampled from a single image.

In addition, we introduce the product manifold to video data for action classification. A novel geometric framework called Graßmann Product Manifolds (GPM) is proposed for video classification in a product space. We represent a video as a data tensor and relate it to a product manifold.

1.5 Contributions

In this section, we summarize the main contributions of this dissertation.

- Formulate visual recognition on special manifolds
- Bridge the gap between a single image and image-set on special manifolds and demonstrate its utility on face recognition
- Exploit image perturbation in registration and illumination spaces
- Apply a statistical illumination model to relight a novel image
- Propose a novel distance measure on Stiefel manifolds
- Illustrate the geometric interpretation for a novelty filter
- Characterize tangent spaces from local neighborhoods

- Use both holistic and local features for face recognition
- Introduce a many to few matching strategy
- Characterize human actions on videos as third order tensors
- Introduce an alternative way for high order singular value decomposition
- Represent a data tensor as a point on a product manifold and illustrate its effectiveness on action classification
- Formulate the geodesic distance on product manifolds
- Emphasize the prominence of the geometry of space and promote novel geometric frameworks on special manifolds

1.6 Overview of Chapters

This dissertation consists of seven chapters. They are organized as follows:

The literature related to special manifolds for computer vision is reviewed in Chapter 2. The topics include special manifolds for optimization, the mutual subspace method and its variants, kernel methods for subspace distances, statistical analysis, variations in distances, and related applications.

Chapter 3 presents the mathematical background on special manifolds. The mathematics covered in this chapter include Stiefel manifolds, Graßmann manifolds, canonical metrics, subspace metrics, gradient flows, and tensor algebra. These materials provides the necessary background for our geometric frameworks and will facilitate the discussions in following chapters.

Chapter 4 models the geometry of illumination spaces using a single image. A statistical illumination model is exploited to relight a single image to a set of illumination variants. A novel distance measure called Canonical Stiefel Quotient (CSQ) is proposed for classification on Stiefel manifolds. CSQ performs two projections on a tangent space and the magnitudes of these projections are measured by the canonical metric. Experimental results on the CMU-PIE and YaleB datasets are described.

Chapter 5 introduces our Graßmann Registration Manifolds (GRM) for face recognition. We illustrate how to characterize a single image on a Graßmann manifold by sampling a registration manifold. The local neighborhood from a registration manifold is extracted to form a tangent space. The tangent spaces are then embedded on a Graßmann manifold and the geodesic distance is computed for discrimination. Experimental results on the FERET datasets are reported.

A new geometric framework called Graßmann Product Manifolds (GPM) is presented for action classification in Chapter 6. This new framework represents a video as a data tensor and characterizes it as an element on a product manifold. Action classification is then performed in this product space. Experiments on the Cambridge-gesture and the KTH-human-action datasets show excellent results.

Finally, the conclusions of this dissertation and possible future research are discussed in Chapter 7.

Chapter 2

Literature Review

In this chapter, we review subjects related to the use of special manifolds for pattern recognition. The specific topics reviewed in this chapter include optimization, mutual subspace methods, kernel methods for subspace distance, statistical analysis, variations in distances, and related applications. Recent survey papers on general topics including single image matching and image-set matching can be found in [102, 122].

2.1 Special Manifolds for Optimization

Since special manifolds are curved spaces, optimization methods defined on vector spaces are not suitable. Edelman et al. [28] formulated some common optimization techniques including the gradient descent, Newton's method, and conjugate gradient method on special manifolds. Later, Manton [71] introduced optimization methods with unitary constraints on special manifolds. More recently optimization methods on matrix manifolds, e.g., trust-region methods have been reported in [1].

In computer vision applications, Ma et al. [68] utilized Newton's method on Stiefel manifolds for estimating 3D motion. The objective function is defined on an essential manifold which is a product of Stiefel manifolds. The natural geometric structure on Stiefel manifolds is considered for 3D motion recovery from image correspondences. The 3D motion estimation on Riemannian manifolds has also been studied by Lee [60].

Subspace estimation is a common task for many statistical and pattern recognition problems [66, 64, 13, 100]. The projection matrix can be obtained from Graßmann manifolds. Liu et

al. [66] employed discriminant analysis for face recognition. An iterative approach is developed for searching the Graßmann manifold for an optimal projection matrix. This method is further extended using simulated annealing to avoid local minima. Lin et al. [64] proposed a Maximum Effective Information (MEI) criterion for face recognition. The MEI criterion essentially seeks a projection matrix maximizing the mutual information. The projection matrix is iteratively searched on a Graßmann manifold using the conjugate gradient method. The conjugate gradient method on Graßmann manifolds has also been exploited for subspace selection [13] where the objective is to maximize the harmonic mean of the symmetric KL divergences between all class pairs.

2.2 The Mutual Subspace Method and its Variants

Yamaguchi et al. [118] first employed canonical angles for face recognition in video sequences in 1998. The authors proposed a method called the Mutual Subspace Method (MSM). The MSM views a video sequence as a set of images and represents it as a linear subspace. The smallest canonical angle between linear subspaces is computed as the distance measure.

The MSM was further enhanced by the Constrained Mutual Subspace Method (CMSM) [34]. To reduce the effect of lighting changes, CMSM first projects image data onto an illumination subspace and applies the MSM to the projected space for classification. Multiple constraints were considered by Nishiyama et al. [79] in addition to MSM. This method employs ensemble learning to form the constraint subspaces, as such the input data are projected onto each constraint subspace followed by MSM. The similarity scores are combined using a weighted sum where the weights are learnt from boosting [32]. The CMSM was later extended to multiple views [36, 69, 70] and incorporated with kernels [35, 37]. The MSM was deployed to a surveillance system to recognize faces from a cluster of moving people [80]. Multiple sets of face images were acquired using multiple cameras.

Recently, CMSM has been extended to multiple image-sets [61]. The prototype is computed as a weighted Karcher mean on a Graßmann manifold. The generalized difference subspace is obtained by computing the difference subspace between each data set and the sum of all

prototypes. The CMSM is then applied to the generalized difference subspace . Furthermore, Adaboost is used to enhance the classification performance.

Li et al. [62] employed the MSM to both holistic and local features for face recognition. Instead of formulating face recognition as a multi-class problem, it is expressed as a two class problem in which a pair of face images belongs either to the same or different people. Canonical angles are computed from both holistic and local features, and boosting is applied to weight the canonical angles.

2.3 Kernel Methods for Subspace Distances

Kernel methods map data to a higher dimensional space, possibly a nonlinear space, through a nonlinear feature map. In 2003, Wolf and Shashua [115] first applied kernel functions to MSM in which the distance function remains positive definite. To do so, a kernelized orthogonalization procedure called kernel Gram-Schmidt was proposed so that the kernelized orthogonal matrices do not need to be evaluated explicitly. In addition, the chosen kernel has proven to be positive definite and is related to Fubini-Study.

Since then, kernel methods have become popular additions for subspace distances. Fukui and Yamaguchi [37] kernelized MSM and applied it to multi-view images. This method employs kernel PCA followed by the kernel whitening transformation. A similar kernelized framework was also applied to CMSM [35] in conjunction with the generalized difference subspace. The generalized difference subspace is characterized as the difference between each subspace and the sum of all subspaces.

Because canonical angles are related to the geodesic distance on Graßmann manifolds, the geodesic distances can also be kernelized. Wang and Shi [113] kernelized geodesic distances including the arc-length and the chordal distance (projection F-norm) for face recognition using image-sets. Nevertheless, the authors found that kernel subspaces (KPCA) followed by the Graßmann discriminant analysis [41] outperformed the kernelized geodesic distances.

Hamm and Lee [41] proposed the projection kernel and Binet-Cauchy kernel for discriminant analysis. In addition, the authors related probabilistic measures to these kernels [40] and showed

that the KL distance in the limit approaches the projection kernel on Graßmann manifolds. In this formulation, data are assumed to be i.i.d. and distributions are modeled as the mixture of factor analyzers. From this observation, the projection kernel was extended to affine and scaled subspaces.

2.4 Statistical Analysis

Many computer vision applications involve machine learning from exemplars. The techniques used in machine learning can be broadly divided into two major frameworks. They are discriminant analysis and regression analysis. The aim of these statistical analyses is to learn a robust projection for pattern recognition.

2.4.1 Discriminant Analysis

Discriminant analysis may be the most widely used framework for pattern recognition. Kim et al. [49] extended the Fisher discriminant analysis from single images to image-sets using Canonical Correlation Analysis (CCA). The distances of within-class and between-class are defined in terms of image-set measured by canonical angles. To broaden the discriminant analysis to nonlinear spaces, Humm and Lee [41] proposed a method called Graßmannian Discriminant Analysis (GDA). This method introduces the projection kernel and Binet-Cauchy kernel in conjunction with Kernel Linear Discriminant Analysis (KLDA).

Fan and Yeung [29] proposed an image-set face recognition algorithm based on local linearity, and the intrapersonal and extrapersonal subspaces. The local linear model is constructed using hierarchical agglomerative clustering. Then, the subspace similarities for intraperson and extraperson are characterized by the largest canonical angle. The difference between the intrapersonal subspace and the extrapersonal subspace is characterized as similarity measure.

2.4.2 Regression Analysis

Pattern recognition problems can also be formulated using regression analysis. Baklr et al. [6] expressed a multivariate regression problem on Stiefel manifolds. The predictor matrix is factorized using singular value decomposition. The free gradients with respect to each variables are

then computed and the gradient descent method is applied to update the projection matrices on a Stiefel manifold. The authors applied this regression to image restoration.

Pham and Venkatesh [82] introduced a multivariate lasso regression for multi-category classification. The loss function of the multivariate lasso regression is a quadratic function in addition to L_1 norm regularization. A suboptimal solution is proposed in which the data projection matrix is found using the steepest descent method on a Stiefel manifold and the regularization matrix is computed by a L_1 solver alternatively.

2.5 Variations in Distances

Just as there are many different ways of formulating distances in Euclidean space, there are many definitions in distances related to canonical angles [28]. For example, one can express the geodesic distance based on the intrinsic geometry of a Graßmann manifold, i.e. the arc-length distance; embedding a Graßmann manifold in the vector space of $\mathbb{R}^{n\times p}$, i.e. the chordal 2-norm and chordal Frobenius-norm distances; the Plücker embedding, i.e. the Fubini-Study metric; or viewing Graßmannian as a projective space in a sphere, i.e. the projection F-norm.

Other heuristics have also been introduced for the utilization of canonical angles. Kim et al. [47] learned a set of weights for canonical angles based on training data. The canonical angles are computed for the holistic image as well as local patches where the local patches are determined using probabilistic PCA. Finally, the holistic and the best local patch are fused by piece-wise linear approximations. Using the canonical angles from both the holistic and local patches has also been discussed [62] in which the weights are learnt from boosting.

Wang et al. [111] proposed the use of weighted average between the variation distance and the exemplar distance as a manifold to manifold distance measure. First, a subset of images from a set of images is selected based on local linearity where the local linearity is defined as the ratio between the geodesic distance (from the adjacency graph) and the Euclidean distance. The reciprocal of the summation of canonical correlations is defined as the variation distance measure and the correlation between the orthogonal exemplar samples is evaluated as the exemplar distance measure. The manifold to manifold distance is finally formulated as the weighted

average of variation and exemplar distances.

Instead of using canonical angles, Wang et al. [110] proposed a new subspace distance for image-sets classification where the distance has later proven to be a distance metric in [101]. This subspace distance is basically computing the summation of correlations between all the basis vectors. The kernel extension of this metric is also given. Chin and Suter [22] formulated a new image-set measure based on a matrix perturbation theorem. It turns out that this new distance measure is the noise term subtracted by the chordal distance where the noise term is related to the ratio between the the sum of singular values from the null space and the sum of singular values from the range space. Recently, Zuccon et al. [125] have introduced a subspace distance formulated by the associated projectors of data matrices. The data matrix of the projector is constructed using co-occurrence statistics. This method considers the projection of a subspace into another rather than the intersection.

Much of the work involving distances defined in terms of canonical angles further complicate matters by selecting among the possible canonical angles. Wolf and Shashua [115] used the first six smallest canonical angles, Fukui and Yamaguchi [36] employed the first three smallest canonical angles, Maeda et al. [69] applied the third smallest canonical angle, Yamaguchi et al. [118] and Beveridge et al. [11] used the smallest canonical angle while Oja and Parkkinen [81], and Fan and Yeung [29] employed the largest canonical angle, and Cheng et al. [18] exercised various truncated angles. Finally, Lui et al. [67] utilized all canonical angles.

2.6 Related Applications

Special manifolds have been exploited in many computer vision applications. The following summarizes these applications, in particular visual tracking, activity and action recognition, and classification in illumination spaces.

2.6.1 Visual Tracking

Subspace methods are often employed for visual tracking. Wang et al. [112] performed online face tracking that utilizes a novelty filter as an evaluation function. The projection matrix at

each time step is found from a Graßmann manifold. The incremental scheme is performed in conjunction with a Kalman filter and factor sampling.

The Mean Shift (MS) algorithm [20] is a nonparametric density estimator that tracks the modes of a distribution, and has been widely employed for visual tracking, segmentation, and clustering. Instead of formulating the MS in Euclidean spaces, Subbarao and Meer [99] formulated the nonlinear counterpart on Graßmann manifolds and Lie groups. This method computes the mean shift as weighted tangent vectors on tangent spaces and maps it back to the manifold via the exponential map.

While the construction of tangent spaces is necessary in [99], Cetingul and Vidal [17] have proposed an alternative method for the nonlinear MS algorithm on Stiefel and Graßmann manifolds. This method avoids the involvement of tangent spaces, in other words, no exponential map is needed. The new nonlinear MS algorithm performs kernel density estimation on the manifold of interests and locates the modes of a distribution intrinsically via iterative optimization.

2.6.2 Activity and Action Recognition

Activity and action classification have received attentions in recent years [75, 106, 84] due in part to the potential applications. These applications include human-computer interaction, automatic annotation of videos, intelligent surveillance, etc. Jia and Yueng [44] introduced a method for local spatio-temporal discriminant embedding. This method finds an optimal embedding that maximizes the canonical angles between temporal subspaces associated with different classes. As such, data points are projected in a local neighborhood where data points of the same action are close while different actions are far apart.

Linear Dynamic Systems (LDS) can be used to identify activities from videos. Saisan et al. [90] employed dynamical systems to model image sequences. The Martin distance depicted by canonical angles is employed to compare the LDS models. Turaga and Chellappa [105] considered activities as LDS governed by a set of system parameters. The parameters of LDS are assumed to be varying with time but locally time-invariant. This locally time-invariant LDS is then modeled as trajectories on Graßmann manifolds and characterized by the Procrustes distance. Statistical inference on special manifolds for activity recognition has also been investi-

gated in [107].

Action videos can generally be considered as a 3rd order tensor. Kim et al. [50] introduced the tensor canonical correlation analysis for action classification. This method seeks projection matrices that maximize the inner product between tensors on each single-shared mode tensor. The similarity between videos is the sum of canonical correlations from all single-shared mode tensors.

2.6.3 Classification in Illumination Spaces

It is known that a set of convex objects under a fixed pose and with a Lambertian reflectance surface forms a vector space. More specifically, this space is called illumination cone. Since illumination cones admit vector space structures, they are elements on Graßmann manifolds. Cheng et al. [18] and Beveridge et al. [11] exploited this principle and performed face recognition in the context of image-sets where the image-sets are a set of face images collected from various illumination variants under the fixed pose assumption. This method was further extended to low resolution in illumination spaces [19].

Chapter 3

Mathematical Background

This chapter reviews the mathematical background on special manifolds. The subjects include special orthogonal groups, Stiefel manifolds, Graßmann manifolds, geodesic flows, and tensor algebra. These materials serve as a vehicle in our research and will facilitate our discussion in following chapters. First, we give the mathematical notations in Table 3.1.

Notations	Descriptions
$\mathcal{A}$	Tensor (calligraphic letters)
A	Matrix (upper-case letters)
a	Vector (lower-case letters)
$\mathcal{M}$	Manifold (a calligraphic letter $\mathcal{M}$)
$T_x\mathcal{M}$	The tangent space to the manifold at $x \in \mathcal{M}$
$\mathbb{SO}(n)$	An $n \times n$ special orthogonal Group (Lie group)
$\mathfrak{so}(n)$	Lie algebra
$\lfloor A \rfloor$	Element on a manifold constructed by a matrix A
$\mathcal{R}(X)$	The range of a data matrix X
$A_{(n)}$	Mode-n flattened matrix
$U^{(n)}$	Mode-n orthonormal matrix
$\times_n$	Mode-n multiplication (Mode-n product)
$I_{n,p}$	An $n \times p$ identity matrix where the bottom $n - p \times p$ submatrix is zero
$0_{n,p}$	An $n \times p$ zero block matrix
$\|\cdot\|$	The Euclidean norm
$\|\cdot\|_F$	The Frobenius norm
tr	The trace of a matrix
X $\times$ Y	Cartesian product of sets or manifolds
$\mathcal{V}_{n,p}$	Stiefel manifold (a set of p-dimensional linear subspaces in $\mathbb{R}^n$)
$\mathcal{G}_{n,p}$	Graßmann manifold (a set of p-dimensional linear subspaces in $\mathbb{R}^n$)
$\otimes$	Kronecker product

Table 3.1: Mathematical Notations

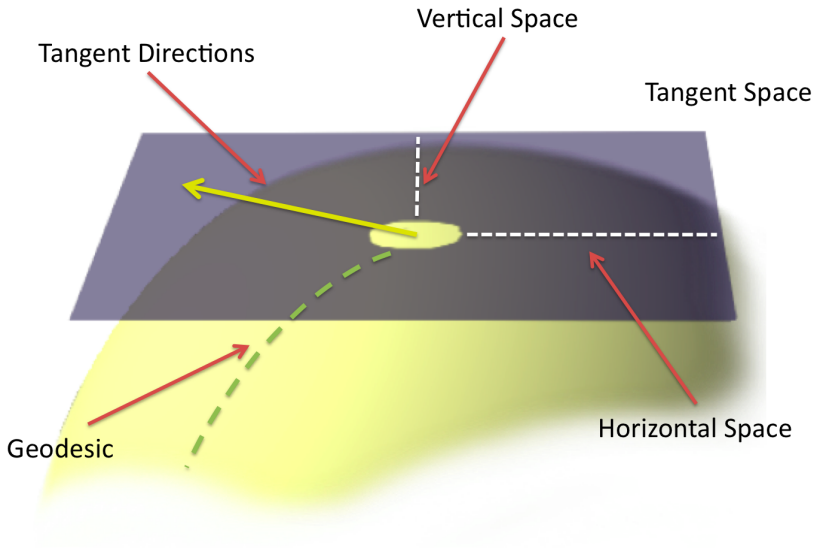


Figure 3.1: Basic geometry of a manifold and its tangent space at a point

The general review materials for manifolds or special manifolds can be found in [28, 71, 58, 21, 1]. Figure 3.1 gives general descriptions of some manifold terminologies. From a differential geometry point of view [58], geodesic distance is the shortest distance between two points on a manifold. Every point on a manifold can form a tangent space. Furthermore, a tangent space can be factorized to a vertical space and horizontal space. When we make a projection on a tangent space, it yields tangent directions. Details of these topics will be discussed in this chapter.

3.1 Lie Groups and Quotient Spaces

Manifolds related to a group structure are specified as Lie groups in which there exists a group operation $\oplus$ such that any element in the group has an inverse and any pair of elements in the group remains in the group after the group operation $X \oplus Y$.

Quotient spaces induced by an equivalence relation define a set of equivalence classes of points in a topological space. This concept allows us to identify the elements of a set to a point through a projection map. In this section, we will review these two important concepts for special manifolds.

3.1.1 Lie Groups

Let $\mathbb{GL}(n)$ be a set of $n \times n$ matrices defined as :

$$\mathbb{GL}(n) = \{X \in \mathbb{R}^{n \times n} : \det(X) \neq 0\} \tag{3.1}$$

Formally, $\mathbb{GL}(n)$ is a set of nonsingular $n \times n$ matrices closed under a group operation, i.e. matrix multiplication. This is because the product of two nonsingular matrices is a nonsingular matrix. This set of matrices is called general linear group. We can further define the orthogonal group and special orthogonal group. Let $\mathbb{O}(n)$ be a set of $n \times n$ orthogonal matrices defined as:

$$\mathbb{O}(n) = \{X \in \mathbb{R}^{n \times n} : X^T X = I\} \tag{3.2}$$

This set of matrices is called orthogonal group. Furthermore, orthogonal matrices could have a determinant either +1 or -1. However, these groups of orthogonal matrices do not connect on a manifold, thus, a subgroup is usually specified. Let $\mathbb{SO}(n)$ be another set of $n \times n$ orthogonal matrices defined as:

$$\mathbb{SO}(n) = \{X \in \mathbb{R}^{n \times n} : X^T X = I, \det(X) = 1\} \tag{3.3}$$

The additional determinant constraint (det = 1) ensures that all matrices in this group are rotation matrices. This set of matrices is called special orthogonal group. The space of $\mathbb{SO}(n)$ at identity is commonly referred to as the Lie group and its tangent space is referred to as the Lie algebra, $\mathfrak{so}(n)$. It can also be seen that $\mathbb{SO}(n) \subset \mathbb{O}(n) \subset \mathbb{GL}(n)$.

3.1.2 Quotient Spaces

Let S be a set and a, b, and $c \in S$. A relation $\sim$ on a set S is called an equivalence relation if it is reflexive ($a \sim a$), symmetric ($a \sim b \rightarrow b \sim a$), and transitive ($a \sim b$ and $b \sim c \rightarrow a \sim c$). For *each* $x \in X$, the equivalence class of x is the set of *all* $y \in X$ such that $x \sim y$.

Let X be a set and $\sim$ be an equivalence relation on X. Then X / $\sim$ is the set of equivalence classes in X. There exists a natural projection, π: $X \rightarrow X$ / $\sim$, mapping a point to its equivalence class. The space X / $\sim$ is called the quotient space of X given by an equivalence relation. In other words, X / $\sim$ is constructed from X by representing the equivalence classes of X to points.

The representation of quotient spaces provides a way to abstract certain elements from a set to a point using an equivalence relation. Points on Stiefel and Graßmann manifolds may be viewed in terms of quotient space representation.

3.2 Stiefel Manifolds

Every data matrix can be orthogonalized, therefore, every image-set can be mapped to an orthonormal matrix $\in \mathbb{R}^{n\times p}$ where n is the dimension of an image and p is the number of images in the set. The space of orthonormal matrices is endowed with specific geometry described as follows. The Stiefel manifold $\mathcal{V}_{n,p}$ is a set of $n \times p$ orthonormal matrices such that

$$\mathcal{V}_{n,p}(Y) = \{Y \in \mathbb{R}^{n\times p} : Y^T Y = I_p\} \tag{3.4}$$

where Y is an element on a Stiefel manifold and I_p is a $p \times p$ identity matrix. This formulation is pictorially described in Figure 3.2. The dimensions of Stiefel manifold $\mathcal{V}_{n,p}$ is np - $\frac{p(p+1)}{2}$ ($\frac{p(p-1)}{2}$ - $p(n-p)$) where $\frac{p(p+1)}{2}$ is the dimension of a normal space. Equation (3.4) reveals that an element on a Stiefel manifold is a collection of images under some mathematical constraints. As such, permuting elements of Y would result at different point on $\mathcal{V}_{n,p}$. Hence, the ordering of the bases is important on $\mathcal{V}_{n,p}$. Furthermore, $\mathcal{V}_{n,p}$ can be viewed as a quotient space of $\mathbb{SO}(n)$ so that we can identify an isotropy subgroup H in $\mathbb{SO}(n)$. H is a set of matrices which does not change Y by the right matrix multiplication. Let Q be an element of $\mathbb{SO}(n)$ and can be factorized into [Y | $Y_\perp$] where $Y \in \mathbb{R}^{n\times p}$ and $Y_\perp \in \mathbb{R}^{n\times(n-p)}$ is the orthogonal complement of Y. Then, the isotropy subgroup H in $\mathbb{SO}(n)$ can be written as:

$$H = \{Z \in \mathbb{SO}(n) : QZ \simeq Q\} \tag{3.5}$$

where $\simeq$ is an equivalence relation that makes the first p columns of a matrix coincide. Then, we can consider a point on a Stiefel manifold as an equivalence class $\lfloor Y \rfloor$ with respect to this equivalence relation described as:

$$\lfloor Y \rfloor = \left\{ [\, Y \,|\, Y_\perp \,] \begin{bmatrix} I_p & 0 \\ 0 & Q_{n-p} \end{bmatrix} : Q_{n-p} \in \mathbb{SO}(n-p) \right\} \tag{3.6}$$

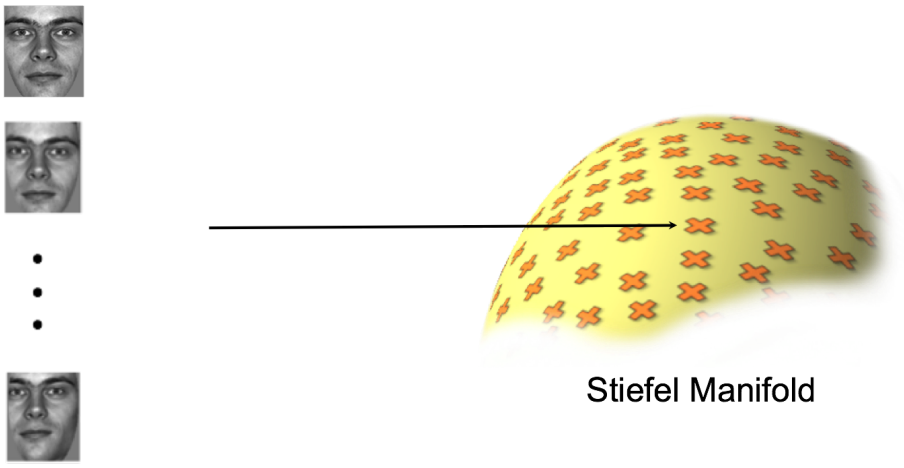


Figure 3.2: A point on a Stiefel manifold

Because the equivalence class only concerns the first p columns of a matrix, a point on a Stiefel manifold is a subset of orthogonal matrices, i.e. $\lfloor Y \rfloor = YI_p$, and that leads to Equation (3.4) as:

$$\lfloor Y \rfloor = \{Y \in \mathbb{R}^{n \times p} : Y^T Y = I_p\} \tag{3.7}$$

Therefore, the Stiefel manifold $\mathcal{V}_{n,p}$ can be viewed as the quotient space defined by $\mathbb{SO}(n)$ / H = $\mathbb{SO}(n)$ / $\mathbb{SO}(n-p)$ under a right matrix multiplication.

3.2.1 Tangent Space

The Stiefel manifold $\mathcal{V}_{n,p}$ may be embedded in a $\mathbb{R}^{n \times p}$ Euclidean space where every element on $\mathcal{V}_{n,p}$ is an orthogonal matrix. Let $Y(t)$ be a matrix in $\mathbb{R}^{n \times p}$ parametrized by a curve t such that $Y(0)$ = $I_{n,p}$ and $Y(t)^T\, Y(t)$ = I. Differentiating $Y(t)^T\, Y(t)$ with respect to t and using the product rule, we have

$$Y(t)^T \frac{d}{dt} Y(t) + \frac{d}{dt} Y(t)^T Y(t) = 0 \tag{3.8}$$

To abbreviate the notation, we let Δ be $\frac{d}{dt}Y(t)$ and drop the parameter t in $Y(t)$, and Equation (3.8) is rewritten as:

$$Y^T \Delta + \Delta^T Y = 0 \tag{3.9}$$

The tangent space at a point Y is a plane tangent embedded on a manifold where the dimension of tangent space is equivalent to the dimension of a manifold $\frac{p(p-1)}{2}$ - $p(n-p)$. For a p

dimensional manifold, the tangent space consists of p tangent vectors describing p directions of travel from the origin at the point Y. Formally, we denote $\mathcal{T}_Y$ the tangent space at Y and express the tangent space as:

$$\mathcal{T}_Y = \{\Delta \in \mathbb{R}^{n\times p} : Y^T\Delta + \Delta^T Y = 0\} \tag{3.10}$$

where $Y^T\ \Delta$ is skew symmetric. To see the skew symmetric property, let $K = Y^T\Delta$, then $\Delta = YK$. Substituting Δ back to Equation (3.9), we have

$$\begin{aligned} Y^T\Delta + \Delta^T Y &= 0 \\ Y^T Y\, K + K^T\, Y^T Y &= 0 \\ K + K^T &= 0 \end{aligned}$$

Thus, $K = Y^T\Delta$ must be skew symmetric.

3.2.2 Normal Space

The tangent space and the normal space are orthogonal complements and orthogonality depends on the choice of an inner product. Considering the inner product on $\mathbb{R}^{n\times p}$ defined by $< M_1, M_2 > = \text{tr} < M_1^T, M_2 >$, we denote $\mathcal{N}_Y$ a normal space at Y and $\mathcal{N} \in \mathcal{N}_Y$. For all Δ in the tangent space and $\mathcal{N}$ in the normal space, we have

$$\text{tr}\{\Delta^T\mathcal{N}\} = 0 \tag{3.11}$$

Furthermore, $\mathcal{N}$ can be written as YS where S is a symmetric matrix. To see that,

$$\text{tr}\{\Delta^T\mathcal{N}\} = \text{tr}\{(YK)^T YS\} = \text{tr}\{K^T Y^T YS\} = \text{tr}\{K^T S\} = 0$$
$$\text{tr}\{(K^T S)^T\} = \text{tr}\{S^T K\} = \text{tr}\{KS^T\} = \text{tr}\{-K^T S^T\} = 0$$

Thus, S is a symmetric matrix. Then, the normal space at Y is defined as:

$$\mathcal{N}_Y = \{\mathcal{N} \in \mathbb{R}^{n\times p} : \mathcal{N} = YS,\ S = S^T\} \tag{3.12}$$

3.2.3 Projection

3.2.3.1 Projection on the Normal Space

We know that a point on the normal space of the Stiefel manifold satisfies $\mathcal{N} = YS$. Let Z be a matrix in $\mathbb{R}^{n\times p}$. A projection of Z onto the normal space at Y is defined as:

$$\Pi_N(Z) = Y \,\text{sym}(Y^T Z) \tag{3.13}$$

where sym(X) = $\frac{(X+X^T)}{2}$. To see Equation (3.13), we have

$$\begin{aligned} Z &= \Delta + \mathcal{N} = \Delta + YS \\ Z^T &= \Delta^T + S^T Y^T = \Delta^T + SY^T \\ Z^T Y &= \Delta^T Y + S \qquad (3.14) \\ Y^T Z &= Y^T \Delta + S \qquad (3.15) \end{aligned}$$

Adding Equation (3.14) and Equation (3.15), and using the fact from Equation (3.9), we have

$$\begin{aligned} (Y^T Z + Z^T Y) &= 2S + \Delta^T Y + Y^T \Delta = 2S \\ S &= \frac{1}{2}(Y^T Z + Z^T Y) = \text{sym}(Y^T Z) \qquad (3.16) \end{aligned}$$

and $\mathcal{N}$ can be written as YS, hence, Equation (3.16) agrees with Equation (3.13).

3.2.3.2 Projection on the Tangent Space

Let Z be a matrix in $\mathbb{R}^{n \times p}$. A projection of Z onto the tangent space at Y is defined as:

$$\begin{aligned} \Delta &= Z - \mathcal{N}_Y \\ \Pi_T(Z) &= Z - \Pi_N(Z) \\ &= Z - Y \,\text{sym}(Y^T Z) \\ &= Z - \frac{1}{2} Y \left(Y^T Z + Z^T Y\right) \\ &= Z - \frac{1}{2} Y \left(Y^T Z + Z^T Y\right) - \frac{1}{2} Y Y^T Z + \frac{1}{2} Y Y^T Z \\ &= \frac{1}{2} Y \left(Y^T Z - Z^T Y\right) + \left(I - Y Y^T\right) Z \\ &= Y \,\text{skew}(Y^T Z) + \left(I - Y Y^T\right) Z \qquad (3.17) \end{aligned}$$

where skew(X) = $\frac{(X-X^T)}{2}$. The tangent directions at Y have the general form defined as:

$$\Delta = YA + Y_\perp B \tag{3.18}$$

3.3 Graßmann Manifolds

A Graßmann manifold $\mathcal{G}_{n,p}$ is a set of p-dimensional linear subspaces of $\mathbb{R}^n$ (p-planes in $\mathbb{R}^n$) for $0 < p \leq n$ where the dimension of $\mathcal{G}_{n,p}$ is $p(n-p)$. Similar to a Stiefel manifold, a Graßmann manifold can also be viewed as a quotient space of $\mathbb{SO}(n)$ and its isotropy subgroup can be represented as:

$$H = \left\{ \begin{bmatrix} Q_p & 0 \\ 0 & Q_{n-p} \end{bmatrix} : Q_p \in \mathbb{SO}(p)\,,\, Q_{n-p} \in \mathbb{SO}(n-p) \right\} \tag{3.19}$$

Section 3.2 shows that a Stiefel manifold can be identified by a quotient representation $\mathcal{V}_{n,p}$ = $\mathbb{SO}(n)$ / $\mathbb{SO}(n-p)$ under a right matrix multiplication by any orthogonal matrix. Let Y be an element on a Stiefel manifold and $Y_\perp$ be its orthogonal complement in $\mathbb{SO}(n)$, then we have

$$[\ Y \mid Y_\perp\ ] \begin{bmatrix} Q_p & 0 \\ 0 & Q_{n-p} \end{bmatrix} = [\ YQ_p \mid Y_\perp Q_{n-p}\ ] \tag{3.20}$$

where $Q_p \in \mathbb{SO}(p)$ and $Q_{n-p} \in \mathbb{SO}(n-p)$. Equation (3.20) reveals the equivalence class wherein the span of the first p columns of Y remains unchanged under a right matrix multiplication, and the right $n-p$ columns have no effect on the span of Y. This is the quotient representation of Graßmann manifolds characterized as $\mathcal{G}_{n,p}$ = $\mathbb{SO}(n)$ / ($\mathbb{SO}(p) \times \mathbb{SO}(n-p)$). Using the quotient representation of Stiefel manifolds, we can represent Graßmann manifolds more concisely as $\mathcal{G}_{n,p}$ = $\mathcal{V}_{n,p}$ / $\mathbb{SO}(p)$. As such, a Graßmannian can be expressed as a homogeneous space [54] which is isomorphic to the quotient space.

The quotient representation of Graßmann manifolds establishes the equivalence relation between orthogonal matrices. That is, two matrices belong to the same equivalence class if their columns span the same p dimensional subspace. Hence, from Equation (3.20), the entire equivalence class can be represented as the subspace spanned by the columns of a given matrix Y.

$$\lfloor Y \rfloor = \{YQ_p : Q_p \in \mathbb{O}(p)\} \tag{3.21}$$

In other words, a point on a Graßmann manifold is a linear subspace which may be specified by any arbitrary orthogonal basis. A Graßmann manifold $\mathcal{G}_{n,p}$ is a collection of all p-dimensional linear subspaces of a Euclidean space in $\mathbb{R}^n$ where the topology of Graßmann manifolds exhibits nonlinear structure. This interpretation is pictorially described in Figure 3.3.

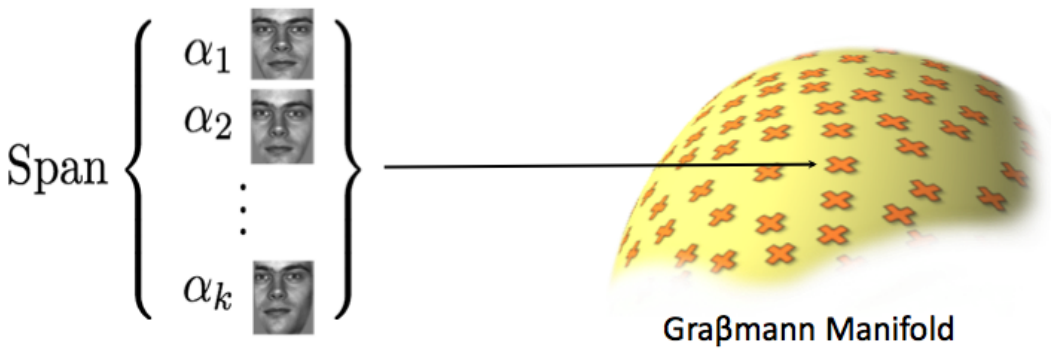


Figure 3.3: A point on a Graßmann manifold

3.3.1 Projection on the Tangent Space of Graßmann Manifolds

Recall that $\Delta = YK$, and K is skew symmetric. Now, let us consider an orthogonal group $\mathbb{O}_n = \{Y \in \mathbb{R}^{n \times n} : Y^T Y = I\}$, then we can express Δ, Y, and K as:

$$Y = [\, Y \mid Y_\perp \,] \tag{3.22}$$

$$\Delta = [\, \Lambda \mid \Lambda_\perp \,] \tag{3.23}$$

$$K = \begin{bmatrix} A & -B^T \\ B & C \end{bmatrix} \tag{3.24}$$

where A is a $p \times p$ skew symmetric matrix, C is a $(n-p) \times (n-p)$ skew symmetric matrix, and B is a $(n-p) \times p$ matrix, and Δ can be rewritten as:

$$\Delta = [\, Y \mid Y_\perp \,] \begin{bmatrix} A & -B^T \\ B & C \end{bmatrix} \tag{3.25}$$

$$= [\, Y\,A + Y_\perp B \mid -YB^T + Y_\perp C \,] \tag{3.26}$$

The tangent space can be decomposed into vertical space and horizontal space where the vertical and horizontal space at a point Y are complementary linear subspaces of the tangent space at Y. Therefore, we can obtain the following:

$$\Delta_V = [\, Y\,A \mid Y_\perp C \,] = Y \begin{bmatrix} A & 0 \\ 0 & C \end{bmatrix} \tag{3.27}$$

$$\Delta_H = [\, Y_\perp B \mid -YB^T \,] = Y \begin{bmatrix} 0 & -B^T \\ B & 0 \end{bmatrix} \tag{3.28}$$

Since the subspace is spanned by the first p columns, that is expressed in Equation (3.18). Furthermore, according to Equation (3.27), we can see that Δ_V does not have any effect on a Graßmann manifold. Thus, the horizontal space Δ_H on a Graßmann manifold is a representation of tangents to the quotient space, hence, the projection on a tangent space on a Graßmann manifold can be written as:

$$\Delta = \Delta_H = Y_\perp B = (I - YY^T)B \tag{3.29}$$

3.4 Canonical Metrics

The canonical metric $g_c : T_y\mathcal{M} \times T_y\mathcal{M} \to \mathbb{R}$ induces an inner product on tangent spaces which allows us to measure the length of tangent vectors at Y. In Euclidean space, we have the standard Euclidean metric defined as:

$$g_e(\Delta, \Delta) = \mathrm{tr}\{\Delta^T\Delta\} \tag{3.30}$$

However, the canonical metric on a Stiefel manifolds is the restriction of the orthogonal group metric defined as:

$$\begin{aligned} g_c(\Delta, \Delta) &= \mathrm{tr}\{\Delta^T\Delta\} \\ &= \mathrm{tr}\{(YA + Y_\perp B)^T(YA + Y_\perp B)\} \\ &= \mathrm{tr}\{A^TA + B^TB\} \\ &= 2\sum_{i\leq j} a_{ij}^2 + \sum_{i=j} b_{ij}^2 \end{aligned} \tag{3.31}$$

Since A is skew symmetric, the Euclidean metric counts the $\frac{p(p+1)}{2}$ independent coordinates of A twice, therefore, we need to divide it by 2. In addition, the canonical metric must be applicable at all points in the Stiefel manifold so that it needs to vary with Y defined as:

$$\begin{aligned} g_c(\Delta, \Delta) &= \frac{1}{2}\mathrm{tr}\{A^TA\} + \mathrm{tr}\{B^TB\} \\ &= \mathrm{tr}\{(YA + Y_\perp B)^T(\frac{1}{2}YA + Y_\perp B)\} \\ &= \mathrm{tr}\{(YA + Y_\perp B)^T(I - \frac{1}{2}YY^T)(YA + Y_\perp B)\} \\ &= \mathrm{tr}\{\Delta^T(I - \frac{1}{2}YY^T)\Delta\} \end{aligned} \tag{3.32}$$

Equation (3.32) is the canonical metric on the Stiefel manifold. On the other hand, $\Delta = Y_\perp$ B is on Graßmann manifolds, and it leads to

$$\begin{aligned} Y^T\Delta &= Y^T Y_\perp B \\ &= Y^T(I - YY^T)B \\ &= 0 \end{aligned} \tag{3.33}$$

Then, the canonical metric on Graßmann manifolds becomes

$$g_c(\Delta, \Delta) = \text{tr}\{\Delta^T(I - \frac{1}{2}YY^T)\Delta\} = \text{tr}\{\Delta^T\Delta\} \tag{3.34}$$

Thus, the canonical metric is equivalent to the Euclidean metric. The canonical metric on Graßmann manifolds can be expressed as:

$$g_c(\Delta, \Delta)_H = \text{tr}\{((I - YY^T)B)^T(I - YY^T)B\} \tag{3.35}$$

where B is an arbitrary $n \times p$ matrix.

3.5 Subspace Metrics

There are several ways to formulate geodesic distances on Graßmann manifolds [28] depending on how the topology of Graßmannian is defined. However, all these geodesic distances are related to canonical angles. In this section, we describe the formulation of computing canonical angles and their geometric interpretation.

3.5.1 Computation of Canonical Angles

Given two data sets X and $Y \in \mathbb{R}^{n\times p}$ (p images embedded in $\mathbb{R}^n$), we seek two projection matrices W_x and $W_y \in \mathbb{R}^{p\times p}$ such that the following objective function [43] is minimized.

$$\min_{W_x, W_y} \parallel XW_x - YW_y \parallel_F^2 \tag{3.36}$$

Note that the special case of Equation (3.36) where W_x and W_y are orthogonal matrices is known as the Procrustes problem [39]. Here W_x and W_y act as rotation matrices such that the rotated data sets XW_x and YW_y are as closely aligned as possible. From a linear algebra point of view,

XW_x can be considered the range of X, similarly, YW_y is the range of Y. Hence, XW_x and YW_y are points on a Graßmann manifold and the minimum distance between these two points is the geodesic distance. When we expand Equation (3.36), we have

$$\mathrm{tr}\left\{W_x^T C_{xx} W_x - 2W_x^T C_{xy} W_y + W_y^T C_{yy} W_y\right\} \tag{3.37}$$

where $C_{xx} = X^T X$, $C_{yy} = Y^T Y$, $C_{xy} = X^T Y$, and $C_{yx} = Y^T X$. Furthermore, we can fix the first and third terms of Equation (3.37) and maximize the second term. As such, Equation (3.37) is expressed as an optimization problem described as follows:

$$\max_{W_x, W_y} \mathrm{tr}\left\{W_x^T C_{xy} W_y\right\} \tag{3.38}$$

subject to

$$\mathrm{tr}\left\{W_x^T C_{xx} W_x\right\} = 1 \tag{3.39}$$

$$\mathrm{tr}\left\{W_y^T C_{yy} W_y\right\} = 1 \tag{3.40}$$

The constraints in Equation (3.38) ensure that points are on special manifolds in which points are ($W_x^T\ C_{xx}\ W_x = (XW_x)^T\ (XW_x) = I$) orthonormal. Since Equation (3.38) is an optimization problem with equality constraints, we express it as Lagrangian formulation:

$$L(W_x, W_y, \lambda_x, \lambda_y) = W_x^T C_{xy} W_y + \lambda_x(1 - W_x^T C_{xx} W_x) + \lambda_y(1 - W_y^T C_{yy} W_y) \tag{3.41}$$

where λ_x and λ_y are the Lagrangian multipliers. The solution can be found by solving the following generalized eigen-system.

$$\begin{bmatrix} 0 & C_{xy} \\ C_{yx} & 0 \end{bmatrix} \begin{bmatrix} W_x \\ W_y \end{bmatrix} = \lambda \begin{bmatrix} C_{xx} & 0 \\ 0 & C_{yy} \end{bmatrix} \begin{bmatrix} W_x \\ W_y \end{bmatrix} \tag{3.42}$$

where λ are the eigenvalues of the generalized eigen-system and are related to canonical angles. The derivation of this solution is given in Appendix A.

3.5.2 Geometric Interpretation of Canonical Angles

While a cosine angle between vectors is often employed to characterize similarity in a Euclidean space, canonical angles, also known as principal angles [43], are used to define a subspace

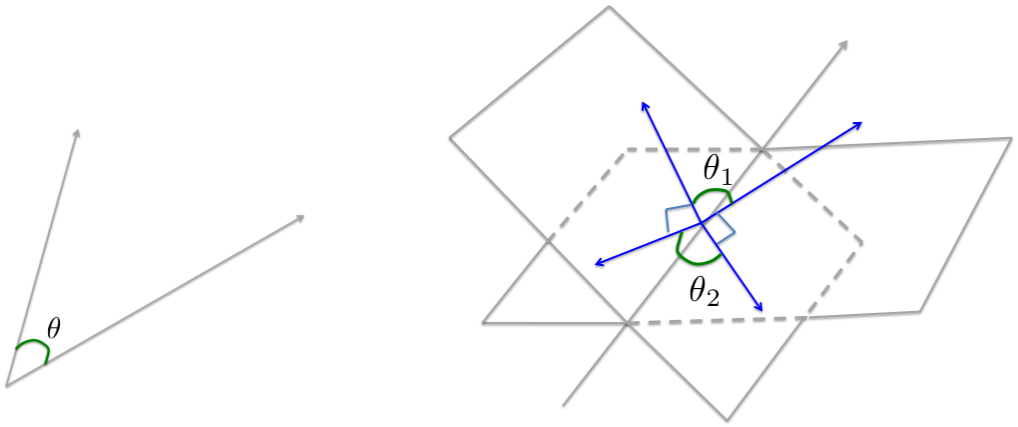


Figure 3.4: Illustration of an angle between vectors and canonical angles between subspaces. An angle between two vectors is a scalar (left) while in general orientation between subspaces is described by a set of angles (right).

distance on a Graßmann manifold. The essential difference is the geometric interpretation. A cosine angle characterized in a single $\mathbb{R}^{n\times p}$ dimension (p = 1) is for vectors while canonical angles are for p-dimensional subspaces of $\mathbb{R}^n$ ($p > 1$). It should be noted that the angle between two vectors is a single angle whereas the canonical angles between two subspaces are a vector of angles. This distinction is illustrated in Figure 3.4 where two planes pass through the origin exhibiting two angles. Figure 3.4 shows two possible canonical angles between two planes. In fact, the optimum θ_1 is zero in this case since two planes are intersected in $\mathbb{R}^3$.

To illustrate the computation of canonical angles between subspaces, let us begin with a definition of an angle between vectors. Given two vectors x and $y \in \mathbb{R}^n$, the angle between these two vectors is defined as:

$$\angle\{x, y\} = \arccos \frac{\langle x, y\rangle}{\| x \| \| y \|} \tag{3.43}$$

For subspaces, one can recursively define a set of angles between them which are called canonical angles. Let $\mathcal{R}(X)$ and $\mathcal{R}(Y)$ be the range of a data matrix X and Y whose dimensions are $n \times p$, respectively. The canonical angles, $\angle_k\{\mathcal{R}(X), \mathcal{R}(Y)\}$, can be recursively defined for $k = 1, \ldots, p$ as:

$$\cos(\theta_k) \quad = \quad \max_{x\in\mathcal{R}(X)} \max_{y\in\mathcal{R}(Y)} x^T y = x_k^T y_k \tag{3.44}$$

subject to

$$\| x \| = \| y \| = 1 \tag{3.45}$$

$$x^T x_i = 0, \qquad y^T y_i = 0, \qquad i = 1, \ldots, k-1 \tag{3.46}$$

Clearly, $\angle_k\{\mathcal{R}(X), \mathcal{R}(Y)\} \in [\ 0,\ \frac{\pi}{2}\ ]$, and Θ is a vector of all canonical angles. When $k = 1$, the constraint given in Equation (3.46) vanishes, otherwise, the computation of canonical angles needs to recursively satisfy the orthogonal constraint shown in Equation (3.46). These canonical angles provide characterization of relative subspace positions, and may be used to compute distances between subspaces. An efficient algorithm for computing canonical angles can be found in [15].

3.5.3 Geodesic Distances

Because the Graßmannian space is curved, distance measures defined in a Euclidean space may not be appropriate. There are several natural metrics to measure the subspace distance on Graßmann manifolds [28]. As the Graßmannian quotient representation illustrates, a subspace distance is invariant under different basis representations. It should also be noted that a subspace distance not only satisfies the metric axioms [58], but also is invariant under any unitary transformation [86].

A geometric structure on a manifold is induced by a metric. Canonical angles between subspaces have several associated metrics. The choice of a subspace metric plays an important role in applications. It is known that the shortest distance between points on a curved space is geodesic. Wong [117] showed that the geodesic distance on a Graßmann manifold is defined using a set of canonical angles shown as:

$$d_g(\mathcal{R}(X), \mathcal{R}(Y)) = \| \Theta \|_2 \tag{3.47}$$

This geodesic distance is also known as arc-length. However, arc-length is not differentiable everywhere. For example, when p = 1, θ_1 is not differentiable at $\frac{\pi}{2}$. This drawback may cause distances falling in the neighborhood of singular points especially in high dimensional cases ($p > 1$). An alternative measure of geodesic distances on Graßmann manifolds is the chordal

distance [23] defined as:

$$d_c(\mathcal{R}(X), \mathcal{R}(Y)) = \| \sin\theta \|_2 \tag{3.48}$$

The chordal distance realizes a projection as a point on the surface of a sphere and the distance is a line segment connecting a point of a sphere to another. Thus, the chordal distance approximates the geodesic distance via a projection embedding and is differentiable everywhere. Bengtsson et al. [10] also pointed out that the chordal distance has advantages over the traditional geodesic distances and the Fubini-Study in mutually unbiased bases, i.e. bases that span the ranges completely orthogonal to each other. In this dissertation, we employ $d_c(\mathcal{R}(X), \mathcal{R}(Y))$ as our subspace metric.

3.6 Gradient Flows on Special Manifolds

The tangent space admits a vector space structure, and often plays an important role in geometric frameworks. Many geometric properties of tangent spaces are derived from gradient flows. This section discusses some related components to tangent spaces including the vertical space, horizontal space, the space of tangent vectors, and the geodesic flows.

Given a criterion function F parametrized by $\{D^{(i)}, L^{(i)}\}_{i=1}^{m}$ where $D^{(i)}$ is a data vector $\in \mathbb{R}^n$, $L^{(i)}$ is a discrete class label. Let U be a projection matrix $\in \mathbb{R}^{n\times p}$ such that a data vector can be projected onto a lower dimensional space, i.e. $U^T D^{(i)}$.

Let Q be an $n \times n$ orthonormal matrix on $\mathbb{SO}(n)$ and J be an $n \times p$ identity matrix such that Q rotates the columns of U to align with the columns of J as follows:

$$Q^T U = J \tag{3.49}$$

$$U = QJ \tag{3.50}$$

Therefore, Q can be factorized as:

$$Q = [U\ V] \tag{3.51}$$

where $V \in \mathbb{R}^{n\times(n-p)}$ such that $U^T V = 0$, $V^T U = 0$, and $V^T V = I$, therefore, V is an orthogonal matrix and is the orthogonal complement of U. In other words, Equation (3.49) can be

rewritten as:

$$Q^T U = \begin{bmatrix} U^T \\ V^T \end{bmatrix} U = \begin{bmatrix} U^T U \\ V^T U \end{bmatrix} = \begin{bmatrix} I_p \\ 0_{(n-p)\times p} \end{bmatrix} = J$$

3.6.1 Geometry of Quotient Spaces

Recall that both Stiefel and Graßmann can be viewed as a quotient space of $\mathbb{SO}(n)$ as such a point on a Stiefel manifold and a Graßmann manifold can be represented as an equivalence class. The quotient geometry of the Stiefel manifold is defined as:

$$\lfloor Q \rfloor = \left\{ Q \begin{bmatrix} I_p & 0 \\ 0 & Q_{(n-p)} \end{bmatrix} : Q_{(n-p)} \in \mathbb{O}(n-p) \right\} \tag{3.52}$$

The quotient geometry of the Graßmann manifold is defined as:

$$\lfloor Q \rfloor = \left\{ Q \begin{bmatrix} Q_p & 0 \\ 0 & Q_{(n-p)} \end{bmatrix} : Q_p \in \mathbb{O}(p) \ , \ Q_{(n-p)} \in \mathbb{O}(n-p) \right\} \tag{3.53}$$

Considering the tangent space at identity such that $Q = I$, the differentials for a Stiefel manifold and a Graßmann manifold are $\begin{bmatrix} 0 & 0 \\ 0 & C \end{bmatrix}$ and $\begin{bmatrix} A & 0 \\ 0 & C \end{bmatrix}$, respectively, where A and C are skew-symmetric matrices.

3.6.2 Vertical Space

The tangent space at Q can be decomposed into vertical and horizontal spaces. The vertical space is defined as a set of tangent vectors of $\lfloor Q \rfloor$. The horizontal space is defined to be the orthogonal complement of the vertical space.

Let A and C be a $p \times p$ and $(n-p) \times (n-p)$ skew-symmetric matrices, respectively. The vertical space on $\mathcal{V}_{n,p}$ is given by

$$\Phi_V = Q \begin{bmatrix} 0 & 0 \\ 0 & C \end{bmatrix} = [U \ V] \begin{bmatrix} 0 & 0 \\ 0 & C \end{bmatrix} = [0 \ \ VC] \tag{3.54}$$

The vertical space on $\mathcal{G}_{n,p}$ is given by

$$\Delta_V = Q \begin{bmatrix} Q(t)A & 0 \\ 0 & C \end{bmatrix} = [U \ V] \begin{bmatrix} A & 0 \\ 0 & C \end{bmatrix} = [UA \ \ VC] \tag{3.55}$$

Both Equation (3.54) and Equation (3.55) demonstrate that the vertical space yields no effect on Stiefel and Graßmann manifolds since the multiplication of UA on Graßmann manifolds has an equivalent class representation to U, and the multiplication of VC is not in the first p columns.

3.6.3 Horizontal Space

Let B be an arbitrary matrix in $\mathbb{R}^{(n-p)\times p}$. Then, the horizontal space on $\mathcal{V}_{n,p}$ is given by

$$\Phi_H = Q \begin{bmatrix} A & -B^T \\ B & 0 \end{bmatrix} = [U\ V] \begin{bmatrix} A & -B^T \\ B & 0 \end{bmatrix} = [U\,A + V\,B \quad -U\,B^T] \tag{3.56}$$

The horizontal space on $\mathcal{G}_{n,p}$ is given by

$$\Delta_H = Q \begin{bmatrix} 0 & -B^T \\ B & 0 \end{bmatrix} = [U\ V] \begin{bmatrix} 0 & -B^T \\ B & 0 \end{bmatrix} = [V\,B \quad -U\,B^T] \tag{3.57}$$

From Equation (3.56) and Equation (3.57), we observe that UB^T is not in the first p columns, thus, the effects of horizontal space on Stiefel and Graßmann manifolds are $UA + VB$ and VB, respectively. They can be further elaborated as:

$$UA + VB = U\text{skew}(A) + (I - UU^T)B \tag{3.58}$$

$$VB = (I - UU^T)B \tag{3.59}$$

which are equivalent to the projections on tangent spaces given in Equation (3.17) and Equation (3.29).

3.6.4 The Space of Tangent Vectors

Due to nonlinearity of special manifolds, it is common to work with a tangent space where it varies from point to point on special manifolds. The space of tangent vectors to a Stiefel manifold $\mathcal{V}_{n,p}$ at U is given by

$$T_U\mathcal{V}_{n,p} = \left\{ Q \begin{bmatrix} A & -B^T \\ B & 0 \end{bmatrix} J \ :\ A = -A^T, A \in \mathbb{R}^{p\times p}, B \in \mathbb{R}^{(n-p)\times p} \right\} \in \mathbb{R}^{n\times p} \tag{3.60}$$

and the space of tangent vectors to a Graßmann manifold $\mathcal{G}_{n,p}$ at U is given by

$$T_U\mathcal{G}_{n,p} = \left\{ Q \begin{bmatrix} 0 & -B^T \\ B & 0 \end{bmatrix} J \ :\ B \in \mathbb{R}^{(n-p)\times p} \right\} \in \mathbb{R}^{n\times p} \tag{3.61}$$

Furthermore, any tangent vector at J [98] can be written as:

$$\begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix} = \sum_{i=1}^{p} \sum_{j=i+1}^{p} \alpha_{ij} E_{ij} \tag{3.62}$$

$$\begin{bmatrix} 0 & -B^T \\ B & 0 \end{bmatrix} = \sum_{i=1}^{p} \sum_{j=p+1}^{n} \alpha_{ij} E_{ij} \tag{3.63}$$

where

$$E_{ij}(k,l) = \begin{cases} 1 & \text{if} \quad k = i \quad \text{and} \quad l = j \\ -1 & \text{if} \quad k = j \quad \text{and} \quad l = i \\ 0 & \text{otherwise} \end{cases}$$

where E_{ij} is an $n \times n$ matrix, and (k, l) is a position of an element of a matrix. To better visualize Equation (3.62) and Equation (3.63), the following $\mathbb{R}^{5\times3}$ example is given for both $\mathcal{V}_{n,p}$ and $\mathcal{G}_{n,p}$ where n is 5 and p is 3.

$$E_{12} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \qquad E_{13} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix}, \qquad E_{23} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}$$

$$E_{14} = \begin{bmatrix} 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad E_{15} = \begin{bmatrix} 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad E_{24} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$E_{25} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}, \quad E_{34} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad E_{35} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

Hence, the summation matrix in $\mathcal{V}_{n,p}$ is given by

$$\begin{aligned} \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & -B^T \\ B & 0 \end{bmatrix} &= \sum_{i=1}^{3} \sum_{j=i+1}^{3} E_{ij} + \sum_{i=1}^{3} \sum_{j=3+1}^{5} E_{ij} \\ &= \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & -1 & -1 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 0 & 1 & 1 & -1 & -1 \\ -1 & 0 & 1 & -1 & -1 \\ -1 & -1 & 0 & -1 & -1 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \end{bmatrix} \end{aligned}$$

The summation matrix in $\mathcal{G}_{n,p}$ is given by

$$\begin{bmatrix} 0 & -B^T \\ B & 0 \end{bmatrix} = \sum_{i=1}^{3} \sum_{j=3+1}^{5} E_{ij} = \begin{bmatrix} 0 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & -1 & -1 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \end{bmatrix}$$

Both of these summation matrices are a skew-symmetric matrix. In addition, Equation (3.60) and Equation (3.61) can be rewritten as:

$$\begin{aligned} T_U \mathcal{V}_{n,p} &= Q \begin{bmatrix} A & -B^T \\ B & 0 \end{bmatrix} J \\ &= Q(\sum_{i=1}^{p} \sum_{j=i+1}^{p} \alpha_{ij} E_{ij} + \sum_{i=1}^{p} \sum_{j=p+1}^{n} \alpha_{ij} E_{ij}) J \qquad (3.64) \\ T_U \mathcal{G}_{n,p} &= Q \begin{bmatrix} 0 & -B^T \\ B & 0 \end{bmatrix} J \\ &= Q(\sum_{i=1}^{p} \sum_{j=p+1}^{n} \alpha_{ij} E_{ij}) J \qquad (3.65) \end{aligned}$$

where

$$\alpha_{ij} = \lim_{\Delta t \downarrow 0} \frac{F(Q \exp(\Delta t E_{ij}) J) - F(U)}{\Delta t} \qquad (3.66)$$

where $\Delta t \downarrow 0$ means that Δt approaches to zero from above since t describes the geodesic and must be positive.

3.6.5 Geodesic Flows on the Orthogonal Group

Since manifolds are curved spaces, the gradient flows must account for its intrinsic geometry. To ensure the movement on a manifold, points are moved along the geodesic curve. Given a tangent space X in a Lie group at Q, a point Q can be moved from $Q(0)$ to $Q(t)$ along the geodesic path expressed as:

$$Q(t) = Q(0) \exp(tX) \qquad (3.67)$$

where $\exp$ is an exponential map given as follows:

$$\begin{aligned} \exp(X) &= I + X + \frac{X^2}{2!} + \frac{X^3}{3!} + \dots \\ &= I + \sum_{k=1}^{\infty} \frac{1}{k!} X^k \qquad (3.68) \end{aligned}$$

To translate a point from a Lie group back to a special manifold, we perform an embedding shown as follows:

$$U = Q \exp(tX) J \tag{3.69}$$

Equation (3.69) illustrates that a point moves along the tangent direction at $Q(0)$ and is mapped back from a Lie algebra to a Lie group. As such, from Equation (3.60), the geodesic path on $\mathcal{V}_{n,p}$ is defined as:

$$U_{(n+1)} = Q_{(n)} \exp\left\{ \Delta t \begin{bmatrix} A & -B^T \\ B & 0 \end{bmatrix} \right\} J \tag{3.70}$$

where n denotes the iteration step. From Equation (3.61), the geodesic path on $\mathcal{G}_{n,p}$ is defined as:

$$U_{(n+1)} = Q_{(n)} \exp\left\{ \Delta t \begin{bmatrix} 0 & -B^T \\ B & 0 \end{bmatrix} \right\} J \tag{3.71}$$

From Equation (3.49), we obtain $U = QJ$. Furthermore, from Equation (3.69), Q, at step $n + 1$, can be updated as follows:

$$\begin{aligned} U_{(n+1)} &= Q_{(n)} \exp(\Delta t\, X) J \\ Q_{(n+1)} J &= Q_{(n)} \exp(\Delta t\, X) J \qquad (3.72) \\ Q_{(n+1)} &= Q_{(n)} \exp(\Delta t\, X) \qquad (3.73) \end{aligned}$$

From Equation (3.72), Q can be updated on $\mathcal{V}_{n,p}$ as:

$$Q_{(n+1)} = Q_{(n)} \exp\left\{ \Delta t \begin{bmatrix} A & -B^T \\ B & 0 \end{bmatrix} \right\} \tag{3.74}$$

and the update of Q on $\mathcal{G}_{n,p}$ can be written as:

$$Q_{(n+1)} = Q_{(n)} \exp\left\{ \Delta t \begin{bmatrix} 0 & -B^T \\ B & 0 \end{bmatrix} \right\} \tag{3.75}$$

Note that the closed forms of A and B depend on the existence of gradient of F. Let the gradient of F be $\frac{dF}{dU} = D \in \mathbb{R}^{n \times p}$, then Equation (3.60) reveals that A and B described in $\mathcal{V}_{n,p}$ can be

computed as:

$$\begin{aligned} D &= T_U \mathcal{V}_{n,p} = Q \begin{bmatrix} A & -B^T \\ B & 0 \end{bmatrix} J = [U\ V] \begin{bmatrix} A & -B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} I_p \\ 0 \end{bmatrix} \\ &= [UA + VB \quad -UB^T] \begin{bmatrix} I_p \\ 0 \end{bmatrix} = VA + VB \qquad (3.76) \\ UA &= D - VB \qquad (3.77) \\ A &= U^T D - U^T V B \qquad (3.78) \\ A &= \frac{U^T D - D^T U}{2} \qquad (3.79) \end{aligned}$$

Since $U^T V = 0$ and A is skew symmetric, $A = U^T D = \frac{U^T D - D^T U}{2}$. Furthermore, we have

$$\begin{aligned} VB &= D - UA \qquad (3.80) \\ B &= V^T D - V^T U A \qquad (3.81) \\ B &= V^T D \qquad (3.82) \end{aligned}$$

Because $V^T U = 0$, $B = V^T D$. In $\mathcal{G}_{n,p}$, B can be computed from Equation (3.61) as:

$$\begin{aligned} D &= T_U \mathcal{G}_{n,p} = Q \begin{bmatrix} 0 & -B^T \\ B & 0 \end{bmatrix} J = [U\ V] \begin{bmatrix} 0 & -B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} I_p \\ 0 \end{bmatrix} \\ &= [VB \quad -UB^T] \begin{bmatrix} I_p \\ 0 \end{bmatrix} = VB \qquad (3.83) \\ B &= V^T D \qquad (3.84) \end{aligned}$$

In the case of non-existence of $\frac{dF}{dU}$, A and B can be numerically obtained from Equation (3.62), Equation (3.63), and Equation (3.66).

3.7 Tensor Algebra

A tensor is high order vector representation, i.e. a vector is a first order tensor and a matrix is a second order tensor, that induces multilinear mappings over a set of vector spaces. Tensor algebra is powerful machinery for analyzing image ensembles. In this section, we review some of the tensor algebra. Recent survey papers of this subject can be found in [53, 2, 26, 85].

3.7.1 The order of Tensors

Let $\mathcal{A}$ be a tensor. The order of a tensor is the number of indices composed the data. For example, the order of a tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ is N. An element of $\mathcal{A}$ is denoted by $a_{i_1 \ldots i_n \ldots i_N}$ where i_n is the index of the order I_n.

3.7.2 Mode k Fibers and Slices

Let $\mathcal{A}$ be a third order tensor. The mode-1 fibers of a third order tensor can be defined as a column vector $\mathcal{A}$(:,j,k). Similarly, slices of a tensor can be defined as a subtensor obtained by fixing the index of one mode, i.e. $\mathcal{A}$(:,:,k).

3.7.3 Matrix Unfolding

The matrix unfolding operation unfolds a tensor to a matrix. It provides a matrix representation of a high-order tensor. The mode-n fibers of $\mathcal{A}$ are the columns of a flattened matrix[1] denoted by $\mathbf{A_{(n)}}$. The flattened matrix $\mathbf{A_{(n)}} \in \mathbb{R}^{I_n \times (I_{n+1} I_{n+2} \ldots I_N I_1 I_2 \ldots I_{n-1})}$ contains the element $a_{i_1 i_2 \ldots i_N}$ at the position with row i_n and column $(i_{n+1}\ i_{n+2}\ \ldots\ i_N\ i_1\ i_2\ \ldots\ i_{n-1})$. For example, the unfolding of a 3rd order tensor $\mathcal{A} \in \mathbb{R}^{I \times J \times K}$ is defined as:

$$\mathbf{A_{(1)}} \in \mathbb{R}^{I \times JK} : a_{ijk} = a^{(1)}_{iv}\ , v = j + (k - 1)K$$
$$\mathbf{A_{(2)}} \in \mathbb{R}^{J \times IK} : a_{ijk} = a^{(2)}_{jv}\ , v = k + (i - 1)I$$
$$\mathbf{A_{(3)}} \in \mathbb{R}^{K \times IJ} : a_{ijk} = a^{(3)}_{kv}\ , v = i + (j - 1)J$$

A pictorial description of the mode-k unfolding from an N order tensor is given in Figure 3.5.

3.7.4 Tensor Matrix Multiplication

The *mode-k product* of a tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ by a matrix $\mathbf{M} \in \mathbb{R}^{J \times I_k}$ is denoted by $\mathcal{A} \times_k \mathbf{M}$ shown as:

$$\mathcal{B} = \mathcal{A} \times_k \mathbf{M} \tag{3.85}$$

[1]The terms, flattened matrix and unfolded matrix, are used interchangeably in this dissertation.

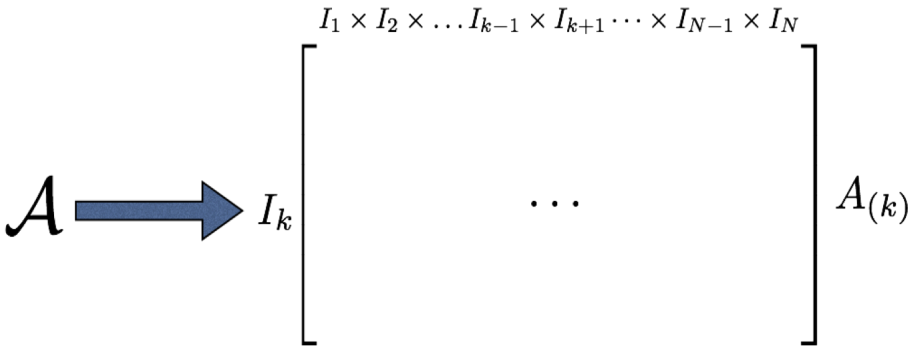


Figure 3.5: The mode-k matrix unfolding from an N order tensor.

where $\mathcal{B} \in \mathbb{R}^{I_1 \times \cdots \times I_{k-1} \times J \times I_{k+1} \times \cdots \times I_N}$ and the entries of $\mathcal{B}$ are computed as:

$$(\mathcal{B})_{i_1 \ldots i_{k-1} i_j i_{k+1} \ldots i_N} = (\mathcal{A} \times_n \mathbf{M})_{i_1 \ldots i_{k-1} i_j i_{k+1} \ldots i_N} = \sum_{k=1}^{I_k} a_{i_1 \ldots i_{k-1} i_k i_{k+1} \ldots i_N} m_{i_j k} \tag{3.86}$$

where $a_{i_1 \ldots i_{k-1} i_k i_{k+1} \ldots i_N}$ is the entry of $\mathcal{A}$, and $m_{i_j k}$ is the entry of $\mathbf{M}$. A pictorial description of the *mode-k product* is given in Figure 3.6. As Figure 3.6 shows, the *mode-k product* consists of three major steps including the mode-k unfolding, matrix the matrix and the mode-k flattened matrix multiplication, and the mode-k folding. We can see that the *mode-k product* can be performed as long as the column dimension of an matrix matches one of the order of a tensor, and the order of the tensor remains unchanged.

In terms of flattened matrices, the *mode-n product* has the following properties.

$$\mathcal{A} \times_m \mathbf{U} \times_n \mathbf{V} = \mathcal{A} \times_n \mathbf{V} \times_m \mathbf{U} \tag{3.87}$$

$$(\mathcal{A} \times_n \mathbf{U}) \times_n \mathbf{V} = \mathcal{A} \times_n (\mathbf{V}\mathbf{U}) \tag{3.88}$$

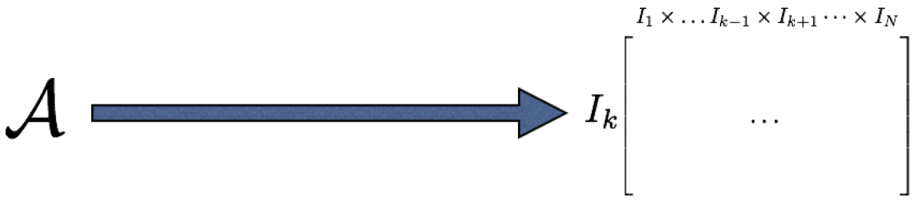
$$\mathcal{B} = \mathcal{A} \times_n \mathbf{M} \Longleftrightarrow \mathbf{B}_{(n)} = \mathbf{M}\mathbf{A}_{(\mathbf{n})} \tag{3.89}$$

Unlike matrices, the order of tensor matrix multiplication can be interchanged shown in Equation (3.87) and Equation (3.88), respectively.

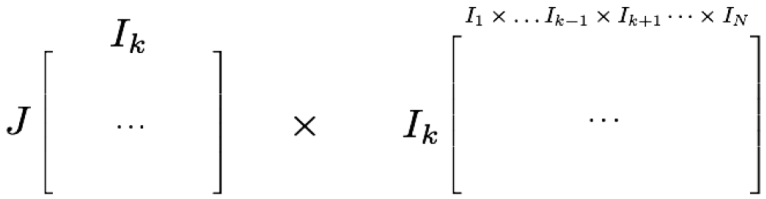
3.8 Tensor Decomposition

A matrix $\mathbf{D} \in \mathbb{R}^{I_1 \times I_2}$ can be decomposed using Singular Value Decomposition (SVD) into the following form:

$$\mathbf{D} = \mathbf{U_1} \mathbf{\Sigma} \mathbf{U_2^T} \tag{3.90}$$



(a) The mode-k unfolding



(b) The matrix and the mode-k flattened matrix multiplication

(c) The mode-k Folding

Figure 3.6: Tensor matrix multiplication

where $\mathbf{U_1} \in \mathbb{R}^{I_1 \times J_1}$, $\mathbf{\Sigma} \in \mathbb{R}^{J_1 \times J_2}$, and $\mathbf{U_2} \in \mathbb{R}^{I_2 \times J_2}$. In terms of the *mode-n product*, it can be rewritten as:

$$\mathbf{D} = \mathbf{\Sigma} \times_{\mathbf{1}} \mathbf{U_1} \times_{\mathbf{2}} \mathbf{U_2} \tag{3.91}$$

Just as a matrix can be factorized using SVD, a tensor can also be factorized using Higher Order Singular Value Decomposition (HOSVD). HOSVD operates on the flattened matrices $A_{(k)}$, and each is factored as follows:

$$A_{(k)} = U^{(k)} \Sigma^{(k)} V^{(k)^T} \tag{3.92}$$

where $\Sigma^{(k)}$ is a diagonal matrix, $U^{(k)}$ is an orthogonal matrix spanning the column space of $A_{(k)}$, and $V^{(k)}$ is an orthogonal matrix spanning the row space of $A_{(k)}$. Then, an N order tensor can be factorized using HOSVD as follows:

$$\mathcal{A} = \mathcal{S} \times_1 U^{(1)} \times_2 U^{(2)} \ldots \times_n U^{(N)} \tag{3.93}$$

where $\mathcal{S} \in \mathbb{R}^{(I_1 \times I_2 \times \cdots \times I_N)}$ is a core tensor, $U^{(1)}, U^{(2)}, \ldots, U^{(N)}$ are orthogonal matrices spanning the column space described in Equation (3.92), and $\times_k$ denotes the mode-k multiplication. The core tensor signifies the interaction of mode matrices and is generally not diagonal when the tensor order is greater than two. Formally, a core tensor $\mathcal{S}$ is iteratively computed as follows:

$$\mathcal{S} = \mathcal{A} \times_1 U^{(1)^T} \times_2 U^{(2)^T} \ldots \times_n U^{(N)^T} \tag{3.94}$$

and the flattened matrix $A_{(n)}$ can be written as:

$$A_{(n)} = U^{(n)} S_{(n)} (U^{(n+1)} \otimes U^{(n+2)} \otimes \ldots U^{(n)} \otimes U^{(1)} \otimes U^{(2)} \ldots U^{(n-1)})^T \tag{3.95}$$

where $\otimes$ denotes the Kronecker product defined as:

$$F \otimes G = (f_{ij}\ G)_{1 \leq i \leq I_1; 1 \leq j \leq I_2} \tag{3.96}$$

where $f_{(ij)}$ is an element of matrix F.

Chapter 4

Canonical Stiefel Quotient

4.1 Introduction

There are several factors commonly associated with the failure of a face recognition algorithm to correctly identify a person. These factors include changes in pose, expression, and illumination. For illumination in particular, it has been argued that changes in illumination can make two images of the same person less similar than two images of different people [76]. In light of this observation, it is challenging to develop a face recognition algorithm which are robust with respect to changes in illumination.

Fortunately, over the last decade, face appearance under varying illumination has been extensively studied and excellent results have been achieved. In particular, the illumination cone principle [9] and the spherical harmonic theory [7]. The illumination cone principle states that a set of images of a convex object with Lambertian reflectance under fixed pose is a convex polyhedral cone in $\mathbb{R}^n$. Any image in the illumination cone can be reconstructed by a linear combination of extreme rays. Moreover, the spherical harmonic theory shows that the Lambertian kernel only contains low frequency components and 99% of the reflected energy can be captured by the first nine spherical harmonics. Many illumination algorithms [38, 45, 59, 95, 121] are derived based on these two fundamental principles.

However a major problem arises when trying to use this theory to build algorithms that are robust to changes in illumination. While the illumination cone for any particular person is relatively stable and well defined, the illumination cone from one person has little in common

with the illumination cone for another person.

This gives rise to an intriguing and important question. *Can a generic illumination model be used to estimate illumination cones for people seen only under one illumination, and if so, are these estimated illumination cones useful for face recognition?* The short answer is yes. We introduce a paradigm for face recognition using illumination spaces, where the illumination spaces are derived based on a generic model built from people other than the person currently being recognized.

Furthermore, we propose a new distance measure called the Canonical Stiefel Quotient (CSQ) for image set classification. While previous methods [7, 38, 45, 95, 121] use illumination models to form a projector from an illumination basis and employ single image matching, we consider the illumination basis as a set of images and perform image set classification.

When we consider image sets for classification, we need to take the underlying geometry into account. We exploit the geometric attributes of special manifolds, namely Stiefel and Graßmann manifolds. These two manifolds often represent natural data in computer vision, especially for image sets in which data can be expressed by orthonormal bases or subspaces.

The reconstructed illumination basis is regarded as a set of relighted illumination variants. With this in mind, we can express every probe and gallery image as a set of relighted illumination variants. From this point forward when considering how to recognize faces, we replace single images with multiple images each representing the person under a different illumination condition. Additionally, every gallery illumination set is represented on a Stiefel manifold where every element is an orthonormal matrix.

To effectively perform image set classification, we first project a probe illumination set on a tangent space of a Stiefel manifold. This projection exhibits tangent directions of the probe illumination variants at a gallery illumination set on a Stiefel manifold, therefore, it can be considered a distance measure when we compute its canonical metric. In addition, we can project the probe illumination variants to the range of a gallery illumination set. This projection becomes another set of illumination variants spanned by the gallery illumination basis. We can then project this new set of illumination variants on the tangent space previously projected. Because

this new set of illumination variants are spanned by the range of the gallery illumination set, this projection represents the tangent directions between a gallery illumination set and its spanning set. Taking the quotient of these two projections on a tangent space with their canonical metrics, we have a new distance measure for classification.

There are several reasons why the proposed method is novel. 1) We relight a single probe / gallery image to a set of illumination variants so that image set matching is possible. 2) We introduce two projections on a tangent space of a Stiefel manifold such that various tangent directions are utilized. 3) The proposed CSQ performs well for generic face recognition. 4) Our CSQ is robust to the choice of training sets.

4.2 Illumination Cone Principle

This section briefly reviews the principle of the illumination cone [9, 7, 87]. The illumination cone can be constructed under two assumptions. First, the object's surface has Lambertian reflectance functions. In other words, there exists a mapping from surface normal to intensities. Second, the shape of the surface of an object is convex.

According to the Lambertian model, an image illuminated by a single point light source at infinity can be expressed as:

$$x = \max(B\, s, 0) \tag{4.1}$$

where x is an image $\in \mathbb{R}^n$, B is the illumination basis $\in \mathbb{R}^{n \times p}$, and $s \in \mathbb{R}^p$ is the lighting coefficient. The $\max$ operator is used to remove all negative components corresponding to the shadowed surface that light sources cannot reach. When the object is illuminated by k light sources at infinity, the image is superposed by the extreme ray given from individual light source described as follows:

$$x = \sum_{i=1}^{k} \max(B\, s_i, 0) \tag{4.2}$$

Equation (4.2) shows that any image in the illumination cone can be reconstructed by a linear combination of extreme rays (images). Furthermore, the space created by the illumination basis

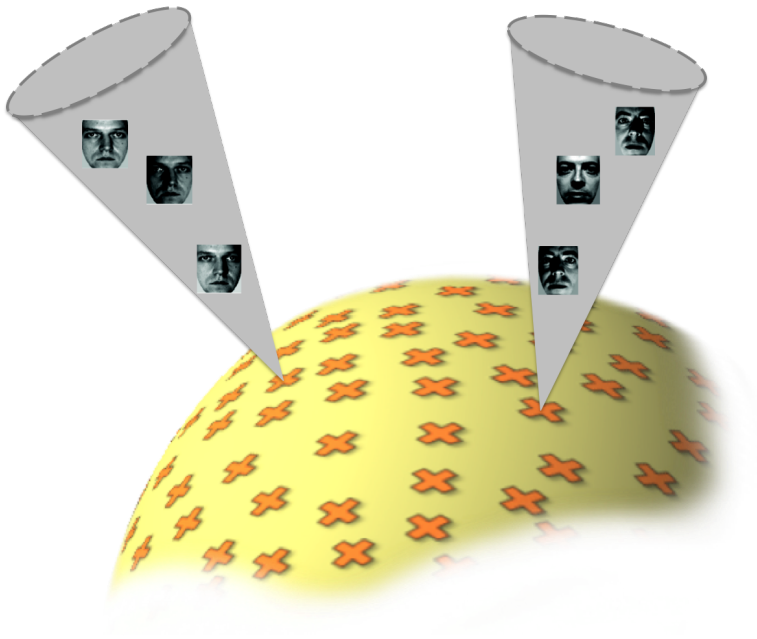


Figure 4.1: Representing illumination cones on a special manifold

B may be expressed as:

$$\mathcal{L} = \{x : x = B\,s\} \tag{4.3}$$

$$\mathcal{L}_0 = \max(\mathcal{L}, 0) \tag{4.4}$$

where $\mathcal{L}_0$ is called the illumination cone or illumination subspace. Belhumeur and Kriegman [9] proved that if both x_1 and x_2 are non-negative, and x_1 and $x_2 \in \mathcal{L}$, then $\lambda x_1 + (1-\lambda)x_2 \in \mathcal{L}_0$ where $\lambda \in [0, 1]$. This means that a set of images in $\mathcal{L}_0$ forms a convex cone in $\mathbb{R}^n$. In addition, Belhumeur and Kriegman showed that an illumination cone can be formed by three linear independent images when no part of the surface is shadowed. Furthermore, Basri and Jacobs [7] showed that a nine dimensional subspace is sufficient to characterize the space of all possible images illuminated from different lighting conditions. Since the illumination cone is constructed from a finite number of extreme rays, it is a polyhedral cone. From a geometric point of view, illumination cones can be conceptualized as points on special manifolds as depicted in Figure 4.1.

Furthermore, the span of illumination spaces can be well-approximated in a low dimensional space. From a spherical harmonics point of view, the relation between a light source and a Lam-

bertian surface can be characterized as convolution on the unit sphere, and therefore represented as multiplication in the frequency domain. In addition, most of the spherical harmonics coefficients are concentrated in the low frequencies.

The illumination cone principle can be summarized as follows:

- A set of images of a convex object with Lambertian reflectance under a fixed pose is a convex polyhedral cone in $\mathbb{R}^n$.
- Any image in the illumination cone can be reconstructed by a linear combination of extreme rays.
- The Lambertain kernel only contains low frequency components and 99% of the reflected energy can be captured by the first nine spherical harmonics.

According to the illumination cone principle, lighting variations can be modeled by approximating the illumination cone. There are two schools of thoughts for modeling the illumination cone. The first is to estimate the illumination basis B in Equation (4.2). The second is to estimate the first nine coefficients from spherical harmonics images. We discuss how to compute the spherical harmonics images in the next subsection.

4.2.1 Spherical Harmonic Images

The spherical harmonics [7] are a set of functions that form an orthonormal basis for the set of all functions on the surface of the sphere. We denote (n_x, n_y, n_z) the surface normal, and λ the vector of the object's albedos such that λ is the albedo of an object point. Furthermore, let n_{x^2}=$n_x n_x$, n_{y^2}=$n_y n_y$, n_{z^2}=$n_z n_z$, n_{xy}=$n_x n_y$, n_{xz}=$n_x n_z$, and n_{xz}=$n_y n_z$. Then, the first nine

harmonic images are defined as [120]:

$$
\begin{aligned}
b_{00} &= \frac{1}{\sqrt{4\pi}}\lambda \\
b_{11}^{e} &= \sqrt{\frac{3}{4\pi}}\lambda \otimes n_x \\
b_{11}^{o} &= \sqrt{\frac{3}{4\pi}}\lambda \otimes n_y \\
b_{10} &= \sqrt{\frac{3}{4\pi}}\lambda \otimes n_z \\
b_{20} &= \frac{1}{2}\sqrt{\frac{5}{4\pi}}\lambda \otimes (2n_{z^2} - n_{x^2} - n_{y^2}) \\
b_{21}^{e} &= 3\sqrt{\frac{5}{12\pi}}\lambda \otimes n_{xz} \\
b_{21}^{o} &= 3\sqrt{\frac{5}{12\pi}}\lambda \otimes n_{yz} \\
b_{22}^{e} &= \frac{3}{2}\sqrt{\frac{5}{12\pi}}\lambda \otimes (n_{x^2} - n_{y^2}) \\
b_{22}^{o} &= 3\sqrt{\frac{5}{12\pi}}\lambda \otimes n_{xy}
\end{aligned}
\tag{4.5}
$$

where $\otimes$ denotes an element-wise multiplication operation, the superscript e and o denote even and odd components of the harmonics, respectively. As Equation (4.5) reveals, the spherical harmonic images are computed based on the 3D shape characterized by the surface normal. The estimation accuracy of the 3D geometry plays a vital role in the spherical harmonic images.

However, there are several concerns when we estimate spherical harmonic images. First, it is difficult to recover the 3D shape from 2D images. Second, there are publicly available illumination datasets such as CMU-PIE [94] or YaleB [38], such that the illumination basis can be directly estimated from these datasets. Third, because human faces are neither entirely Lambertian nor completely convex, spherical harmonic images may not be adequate to model specular reflection, inter-reflection, or cast shadows. When we estimate the nine point basis images from real images, these basis images already contain all the complicated reflections. Therefore, we estimate the illumination basis from a set of real images discussed in the next section.

4.3 Illumination Model

The illumination cone principle states that any image in the illumination cone can be reconstructed by a linear combination of extreme rays defined in Equation (4.2). To approximate an illumination cone, all we need is to estimate the illumination basis, B, for each person. However, human faces are not completely Lambertian and convex.

To overcome the non-convex and non-Lambertian assumptions, Sim and Kanade [95] introduced an error term to the standard Lambertian equation and used a Bayesian method to model the variations. We adopt this statistical illumination model for generic faces whose identities are distinct between a training set and a test set. First, let us describe how we construct the illumination model.

To make the illumination model generic, the augmented Lambertian equation that we use is described as follows:

$$x = Bs + e(s) \tag{4.6}$$

where $e(s)$ is the error term. While Sim and Kanade [95] applied this error term to model shadows and specular reflections, we use this error term to model the difference between a training image and an average image under a specific lighting condition.

4.3.1 The Bayesian Model

A Bayesian framework is employed to formulate the illumination model. For a novel image $\hat{x}$, its new illumination basis can be estimated using the maximum a posterior (MAP) estimate.

$$B^* = \underset{B}{\operatorname{argmax}}\, P(B|\hat{x}) \propto \underset{B}{\operatorname{argmax}}\, P(\hat{x}|B)P(B) \tag{4.7}$$

We can further assume that the illumination basis B is Gaussian distributed. Then, we have

$$P(B) = \frac{1}{\sqrt{(2\pi)^M |\sum|}} \exp(-\frac{1}{2}(B-\mu_B)C_B^{-1}(B-\mu_B)^T) \tag{4.8}$$

$$P(\hat{x}|B) = \frac{1}{\sigma_e(\hat{s})\sqrt{(2\pi)}} \exp(-\frac{1}{2}\frac{\parallel \hat{x}-x \parallel^2}{(\sigma_e(\hat{s}))^2}) \tag{4.9}$$

Using Equation (4.6), we have

$$P(\hat{x}|B) = \frac{1}{\sigma_e(\hat{s})\sqrt{(2\pi)}} \exp(-\frac{1}{2}\frac{\parallel \hat{x}-B\hat{s}-\mu_e(\hat{s}) \parallel^2}{(\sigma_e(\hat{s}))^2}) \tag{4.10}$$

where B is the unknown basis for the probe image $\hat{x}$, μ_B is the mean basis, and C_B is the covariance basis matrix from the training set. After dropping the constant terms and taking the logarithm of the expressions, we have

$$P(B) = -\frac{1}{2}(B-\mu_B)C_B^{-1}(B-\mu_B)^T \tag{4.11}$$

$$P(\hat{x}|B) = -\frac{1}{2}\frac{\parallel \hat{x} - B\hat{s} - \mu_e(\hat{s}) \parallel^2}{(\sigma_e(\hat{s}))^2} \tag{4.12}$$

The mean μ_B and the covariance matrix C from Equation (4.11) can be easily estimated from the training data. All we need to estimate from Equation (4.12) are the estimated lighting coefficients $\hat{s}$, the mean error μ_e given $\hat{s}$, and the standard deviation error σ_e given $\hat{s}$. The following two subsections show how these parameters can be obtained.

4.3.2 Lighting Coefficient Estimation

We denote $s_k^{(p)}$ the lighting coefficient for the subject p under the lighting condition k. Given an image x, the Lambertian equation becomes the following:

$$x_k^{(p)} = B^{(p)} s_k^{(p)} \tag{4.13}$$

where $x_k^{(p)}$ is a training image for subject p under a lighting condition k. The lighting coefficient for the training image $x_k^{(p)}$, can be computed as:

$$s_k^{(p)} = (B^{(p)})^{-1} x_k^{(p)} \tag{4.14}$$

In addition, we assume that we have N training subjects and each subject has M images under different lighting conditions. Let $\hat{x}$ be a novel image, the associated lighting coefficient $\hat{s}$ can be estimated by applying a kernel regression shown as follows:

$$\hat{s} = \frac{\sum_{p=1}^{N}\sum_{k=1}^{M}\alpha_{p,k}s_k^{(p)}}{\sum_{p=1}^{N}\sum_{k=1}^{M}\alpha_{p,k}} \tag{4.15}$$

$$\alpha_{p,k} = \exp(-\frac{\parallel \hat{x} - x_k^{(p)} \parallel^2}{2(\sigma_k^{(p)})^2}) \tag{4.16}$$

and $\sigma_k^{(p)}$ is defined as:

$$\bar{x}_k = \frac{1}{NM}\sum_{p=1}^{N}\sum_{k=1}^{M} x_k^{(p)} \tag{4.17}$$

$$\sigma_k^{(p)} = \sqrt{\frac{1}{NM}\sum_{p=1}^{N}\sum_{k=1}^{M}(x_k^{(p)} - \bar{x}_k)^2} \tag{4.18}$$

4.3.3 Error Term Estimation

The average lighting coefficient over all training subjects under the lighting condition k can be computed as follows:

$$\bar{s}_k = \frac{1}{N}\sum_{p=1}^{N} s_k^{(p)} \tag{4.19}$$

$$\bar{\sigma}_k = \sqrt{\frac{1}{N}\sum_{p=1}^{N}(s_k^{(p)} - \bar{s}_k)^2} \tag{4.20}$$

where N is the number of subjects in the training set. Note that we have $[\bar{s}_1 \mid \bar{s}_2 \mid \ldots \mid \bar{s}_M]$ and $[\sigma_1 \mid \sigma_2 \mid \ldots \mid \sigma_M]$ in the training set. Putting the $\bar{s}_k$ into Equation (4.6), we can obtain the error term for subject p under the lighting condition k as follows:

$$e_p(\bar{s}_k) = x_k^{(p)} - B^{(p)}\bar{s}_k \tag{4.21}$$

This equation expresses the error term for a training image when we use the average lighting coefficient $\bar{s}_k$ over all subjects. The average lighting coefficient $e_p(\bar{s}_k)$ can be varied in terms of p or k. Then, we compute the average error term given the average lighting coefficient $\bar{s}_k$ shown as:

$$\mu_e(\bar{s}_k) = \frac{\sum_{p=1}^{N}\beta_p e_p(\bar{s}_k)}{\sum_{p=1}^{N}\beta_p} \tag{4.22}$$

$$\beta_p = \exp(-\frac{\parallel s_k^{(p)} - \bar{s}_k \parallel^2}{2(\bar{\sigma}_k)^2}) \tag{4.23}$$

where $\bar{s}_k$ and $\bar{\sigma}_k$ are defined in Equation (4.19) and Equation (4.20), respectively. The estimated standard deviation error is calculated as:

$$\sigma_e(\bar{s}_k) = \sqrt{\frac{1}{N}\sum_{p=1}^{N}(e_p(\bar{s}_k) - \mu_e(\bar{s}_k))^2} \tag{4.24}$$

Note that we have estimated $[\mu_e(\bar{s}_1) \mid \mu_e(\bar{s}_2) \mid \ldots \mid \mu_e(\bar{s}_M)]$ and $[\sigma_e(\bar{s}_1) \mid \sigma_e(\bar{s}_2) \mid \ldots \mid \sigma_e(\bar{s}_M)]$.

Using the error term $\mu_e(\bar{s}_k)$ from Equation (4.22), we can apply kernel regression to calculate the mean and standard deviation expressed as follows:

$$\mu_e(\hat{s}) = \frac{\sum_{k=1}^{M} \gamma_k \mu_e(\bar{s}_k)}{\sum_{k=1}^{M} \gamma_k} \tag{4.25}$$

$$\gamma_k = \exp(-\frac{\parallel \hat{s} - \bar{s}_k \parallel^2}{2(\bar{\sigma}_k)^2}) \tag{4.26}$$

$$\sigma_e(\hat{s}) = \sqrt{\frac{1}{M} \sum_{k=1}^{M} (\sigma_e(\bar{s}_k) - \mu_e(\hat{s}))^2} \tag{4.27}$$

where $\hat{s}$ is $[\hat{s}_1 \mid \hat{s}_2 \mid \ldots \mid \hat{s}_M]$.

4.3.4 Illumination Basis Estimation

While traditional MAP estimate determines which illumination basis B has the highest probability, this statistical model seeks a closed-form solution of Equation (4.7) expressed as follows:

$$J = -\frac{1}{2} \frac{\parallel \hat{x} - B\hat{s} - \mu_e \parallel^2}{(\sigma_e)^2} - \frac{1}{2}(B - \mu_B) C_B^{-1} (B - \mu_B)^T \tag{4.28}$$

Taking the derivative of Equation (4.28), we have

$$\frac{\partial J}{\partial B} = \frac{-1}{(\sigma_e)^2}(\hat{x} - B\hat{s} - \mu_e)\hat{s}^T + (B - \mu_B) C_B^{-1} \tag{4.29}$$

Setting $\frac{\partial J}{\partial B}$ = 0, we have

$$B = (\frac{\hat{s}\hat{s}^T}{(\sigma_e)^2} + C_B^{-1})^{-1} (\frac{\hat{x} - \mu_e}{(\sigma_e)^2}\hat{s} + C_B^{-1}\mu_e) \tag{4.30}$$

Using the Woodbury's identity[1], we have

$$B = (\frac{\hat{x} - \mu_B\hat{s} - \mu_e}{(\sigma_e)^2 + \hat{s}^T C_B \hat{s}}) C_B \hat{s} + \mu_B \tag{4.31}$$

As Equation (4.31) reveals, the estimated basis is composed with the mean basis μ_B and the characteristic term $(\frac{\hat{x} - \mu_B\hat{s} - \mu_e}{(\sigma_e)^2 + \hat{s}^T C_B \hat{s}} C_B \hat{s})$. The numerator of the characteristic term, $(\hat{x} - \mu_B\hat{s} - \mu_e)$, describes the difference between the probe image and the reconstructed image using the mean basis images. Equation (4.31) is pictorially shown in Figure 4.2

[1] $(A + UCV)^{-1} = A^{-1}U(C^{-1} - VA^{-1}U)^{-1}VA^{-1}$

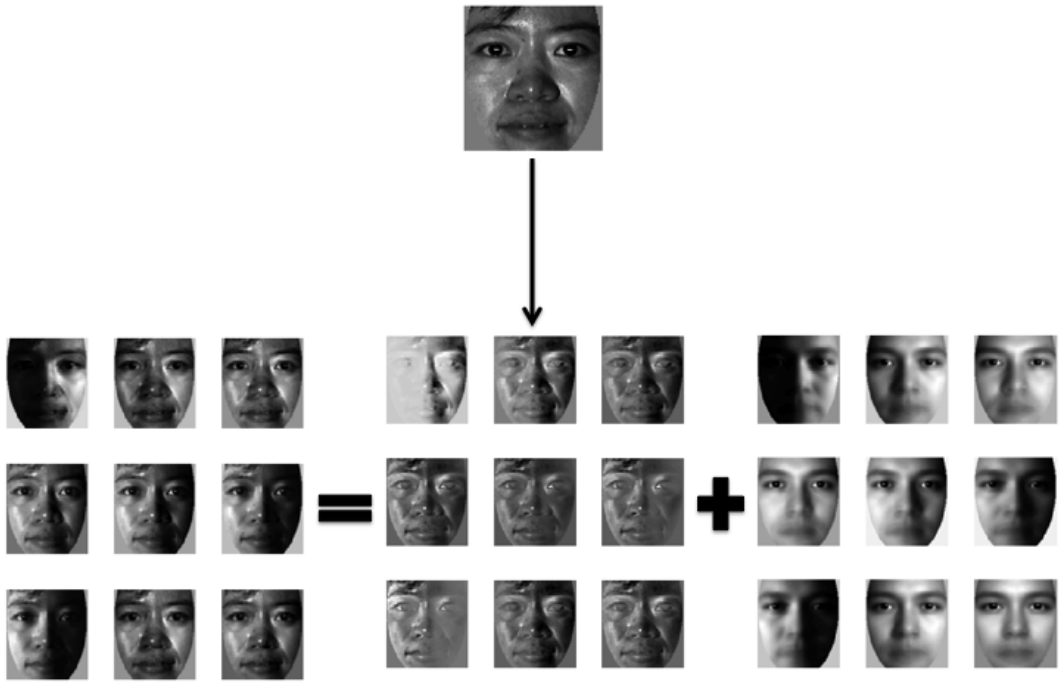


Figure 4.2: Image relighting using a statistical illumination modal

As Figure 4.2 depicts, the relighted images are the combination between the mean images obtained from the training set and the characteristic images. The characteristic image indicates the weight between the novel image and the mean image associated with the variation from the training set. As a result, the illumination model accounts for the person specific information from the novel image and the general illumination from training data.

4.4 Illumination Basis Selection

Before building an illumination basis, we need to determine the lighting configuration. So long as our training data is sufficiently rich in lighting variations that ii effectively spans the entire illumination cone, we may select the nine point subspace [7] and adopt nine illumination images as our illumination basis. These nine images represent the illumination directions that are reconstructed for the probe and gallery images.

When the number of available illumination directions is more then nine, we have the freedom to select the illumination basis. It is known that the Karhunen-Loève basis captures more

statistical variance than any other basis and minimizes the Shannon's entropy [51]. The truncation error of the Karhunen-Loève basis can be computed using the eigenvalues of a covariance matrix. We use a backward selection to choose the illumination basis that has the maximum of total energy. This operation can be expressed as follows:

$$B^*_{n-1} = B^*_n \setminus E_k \tag{4.32}$$

$$E_k = \underset{k \in 1 \ldots |B^*_n|}{\operatorname{argmax}} \sum_{i=1}^{n-1} \lambda_i \tag{4.33}$$

where $\setminus$ is the set difference operator, n is the number of illumination images in the current set, and λ is the eigenvalues of the illumination covariance matrix excluding the k^{th} illumination image where the illumination image is the average image over all training subjects under the specific illumination direction. Thus, Equation (4.32) is a recursive process and is terminated when n reaches nine and $\{B^*_1, B^*_2, \ldots, B^*_9\}$ are the selected illumination directions.

4.5 Canonical Stiefel Quotient

Traditional methods [7, 45, 95, 121] project a novel image to the reconstructed illumination basis and compute the Euclidean distance between the novel image and the projected subspace. These methods do not account for the underlying geometry of illumination spaces. An alternate approach is to use the reconstructed illumination basis as a feature representation and employ the entire illumination cone for classification. This alternative approach takes advantage of the idiosyncratic aspects of illumination cones and undertakes classification using multiple images instead of a single image.

One straight forward approach to performing image-set comparison is to project the illumination basis on a Graßmann manifold and compute the geodesic distance [67]. However, a geodesic method is limited to matching spanning sets, and discards the specific order of the illumination basis which may be vital for classification. Alternatively, one can explore the use of tangent directions on a Stiefel manifold between illumination variants. This is the proposed approach, and it proceeds as follows.

First, all illumination bases for probe and gallery images are reconstructed using Equa-

tion (4.31). Let $Z \in \mathbb{R}^{n\times p}$ be a probe illumination basis and $Y \in \mathbb{R}^{n\times p}$ be an orthonormal gallery basis which represents a point on a Stiefel manifold. Since the geodesic requires retraction on a Stiefel manifold [28] and no differentiable function is available [30], there is no closed-form solution for computing a geodesic distance. We therefore turn our attention to its tangent space. When we project Z on a tangent space at Y of a Stiefel manifold, we have the tangent directions between Z and Y. The length of the tangent directions can be computed using the associated canonical metric. According to Equation (3.18) and Equation (3.32), the tangent directions and the canonical metric on a Stiefel manifold are expressed as:

$$\begin{aligned} \Delta = \Pi_T(Z) &= Y \,\text{skew}(Y^T Z) + (I - YY^T)\, Z \\ g_c(\Delta,\Delta)_S &= \text{tr}\{\Delta^T (I - \frac{1}{2} YY^T)\Delta\} \end{aligned}$$

The canonical metric induces the inner product of tangent directions on a Stiefel manifold, thus it can be used as a distance measure.

Additionally, we can project Z to the range of Y and let Q be $YY^T Z$. When we project Q on a tangent space at Y of a Stiefel manifold, we have

$$\begin{aligned} \Lambda = \Pi_T(Q) &= Y \,\text{skew}(Y^T Q) + (I - YY^T)Q \\ &= Y \,\text{skew}(Y^T Z) \end{aligned} \tag{4.34}$$

and its canonical metric becomes

$$\begin{aligned} g_c(\Lambda,\Lambda)_Q &= \text{tr}\,\{\Lambda^T (I - \frac{1}{2} YY^T)\Lambda\} \\ &= \frac{1}{2}\,\text{tr}\,\{Z^T Y \,\text{skew}(Y^T Z)\} \end{aligned} \tag{4.35}$$

Because Q is in the spanning set of Y, the canonical metric $g_c(\Lambda,\Lambda)_Q$ describes the length of tangent directions between Y and the spanning set of Y on a Stiefel manifold. A new distance measure is inspired by using these two projections on a tangent space defined as:

$$\begin{aligned} CSQ(Z,Y) &:= \frac{g_c(\Delta,\Delta)_S}{\frac{1}{2} + g_c(\Lambda,\Lambda)_Q} \\ &= \frac{2\; g_c(\Delta,\Delta)_S}{1 + \text{tr}\,\{Z^T Y \,\text{skew}(Y^T Z)\}} \end{aligned} \tag{4.36}$$

The $\frac{1}{2}$ augments the denominator to prevent an ill-defined condition[2]. Since we compute the canonical metric for the projections on a Stiefel manifold, and use the quotient for classification, we call this measure the **Canonical Stiefel Quotient** (CSQ). The recognition process is performed as follows:

$$k^* = \underset{k \in gallery}{\operatorname{argmin}} \left\{ \frac{2\ g_c(\Delta_k, \Delta_k)_S}{1 + \operatorname{tr}\left\{Z^T Y_k\ \operatorname{skew}(Y_k^T Z)\right\}} \right\} \tag{4.37}$$

where $\Delta_k = \Pi_T(Z)_k$ is defined in Equation (3.17) and k^* is the identity of the probe subject.

Similarly, one can project Z onto a tangent space of a Graßmann manifold which is equivalent to projecting Z onto its horizontal space. According to Equation (3.29), the tangent directions on a Graßmann manifold are formulated as:

$$\nabla = \Pi_H(Z) = (I - YY^T)Z$$

and the canonical metric on a Graßmann manifold is equivalent to the Euclidean metric defined as:

$$\begin{aligned} g_c(\nabla, \nabla)_H &= \operatorname{tr}\left\{\nabla^T \nabla\right\} \\ &= \operatorname{tr}\left\{((I - YY^T)Z)^T (I - YY^T)Z\right\} \end{aligned} \tag{4.38}$$

It is easy to see that the projection Q (YY^TZ) at Y on the horizontal space of a Graßmann manifold is zero. Therefore, there is no canonical Graßmann quotient. It is worth mentioning that Kohonen and Oja discovered a special case of Equation (4.38) known as the novelty filter [52] where the dimension p is equal to one. The geometric interpretation for the general case ($p \geq 1$) is a projection onto the horizontal space of a Graßmann manifold. We call this general formulation the **Canonical Graßmann Horizontal Projection** (CGHP). The CGHP measures the length of tangent directions on a tangent plane of a Graßmann manifold and can be employed for classification.

---

[2]An ill-define condition, $\frac{0}{0}$, occurs when $Z = Y$, i.e., when the probe image is identical to the gallery image. In practice, this instance is never encountered, indeed, the denominator was actually never close to zero in our experiments.

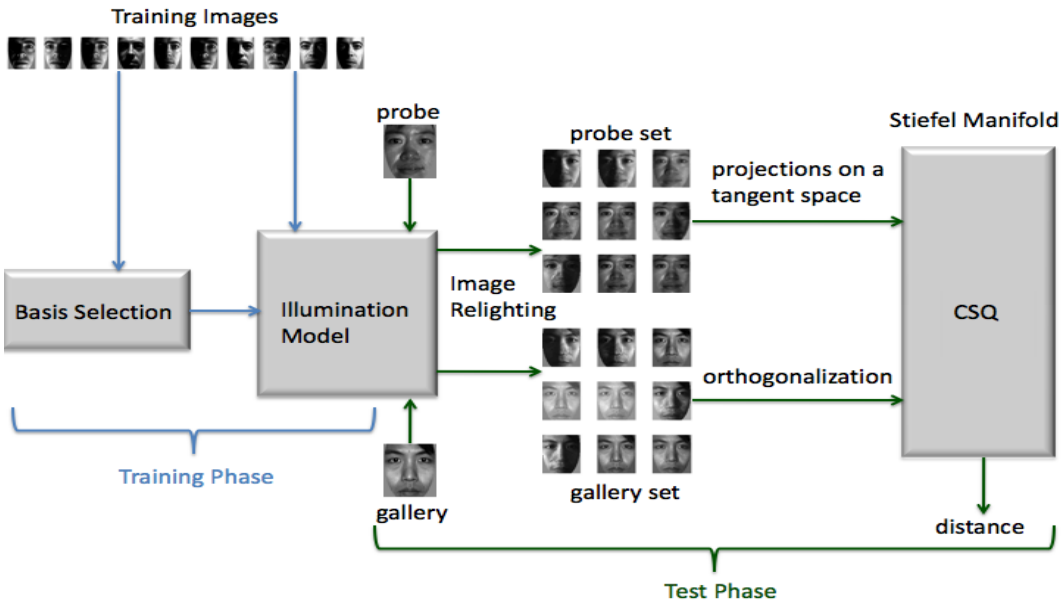


Figure 4.3: The proposed CSQ algorithm where the blue color represents the training phase and the green color represents the test phase.

4.6 Algorithm Summary

To summarize our CSQ algorithm, a system diagram is depicted in Fig. 4.3 where the blue color represents the training phase and the green color represents the test phase. In the training phase, a set of training images is employed to determine a set of illumination directions as described in Section 4.4. Then, the same set of training images and the selected illumination directions are used to build an illumination model described in Section 4.3. At this point, the illumination model is constructed and may be used to relight a set of images from a novel image.

In the test phase, a single probe image and a gallery image are compared. The comparison procedure is outlined as follows. First, both the probe and the gallery images are relighted to a set of probe images and gallery images using the trained illumination model. Then, the gallery set is orthogonalized such that it is represented as a point on a Stiefel manifold. The probe set is projected on the tangent space at a given gallery set on the Stiefel manifold. The projections are carried out using Equation (3.17) and Equation (4.34). Finally, the CSQ is computed as a distance between the probe image and the gallery image using Equation (4.36). In summary, our

Figure 4.4: Example images of PIE database (light on)

CSQ matches image-sets on a Stiefel manifold through a set of relighted images created by an illumination model.

4.7 Experiments

This section describes the data sets, experiment design, baseline algorithms, experimental results, and our findings.

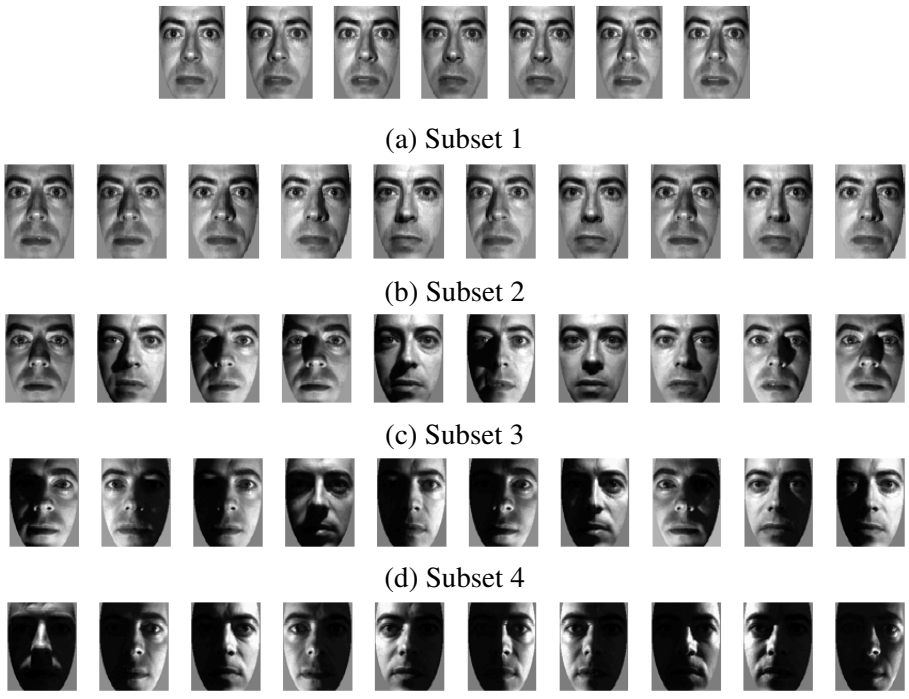
4.7.1 Data Sets

The CMU-PIE [94], YaleB [38], and Extended-YaleB [38] are popular public illumination databases. They have been widely used to evaluate face recognition algorithms. In our experiments, we adopt these databases and assess our algorithm using the frontal pose images (pose 27 for CMU-PIE and pose 0 for YaleB and Extended-YaleB). Examples of these images are shown in Figure 4.4, Figure 4.5, and Figure 4.6. First, we will briefly describe these data sets.

4.7.1.1 CMU-PIE

The CMU-PIE data set consists of 68 subjects and each subject has 48 illumination variants. Furthermore, 24 illumination variants are sampled with the room lights on and the other 24

Figure 4.5: Example images of PIE database (light off)



(a) Subset 1

(b) Subset 2

(c) Subset 3

(d) Subset 4

(d) Subset 5

Figure 4.6: Example images of YaleB database

illumination variants are sampled with the room lights off[3]. All experiments here use the more extreme room lights off images.

4.7.1.2 YaleB and Extended-YaleB

The YaleB data set consists of 10 subjects and each subject has 64 illumination variants. In addition, these 64 illumination variants are further divided into five subsets where the numbers of illumination samples are 7, 12, 12, 14, and 19 for subsets 1 through 5 respectively. The Extended-YaleB data set has the same configuration as the YaleB data set with additional 28 subjects.

4.7.2 Experiment Design

In this work, we are interested in generic face recognition such that the subject identity between training images and test images do not overlap. In addition, we adopt cross-database configuration between training and test sets. The following four protocols are used in our experiments.

I. *Train on the CMU-PIE and test on the CMU-PIE* . We divide the CMU-PIE data set into four partitions, additionally, we train on one partition and test on the other three partitions. As such, 17 subjects are employed for training and 51 subjects are used for testing.

II. *Train on the Extended-YaleB (subset 3 + subset 4 + subset 5) and test on the CMU-PIE* . We use the Extended-YaleB data set for training and test on the entire CMU-PIE data set.

III. *Train on the Extended-YaleB (subset 3 + subset 4 + subset 5) and test on the YaleB.* The Extended-YaleB data set is employed for training and the YaleB data set is used for testing.

IV. *Train on the CMU-PIE and test on the YaleB+Extended-YaleB.* We use 68 subjects from the CMU-PIE data set for training, and combine the YaleB and Extended-YaleB data sets for testing. Thus, the test data set has 38 subjects.

---

[3]There are three illumination variants (00, 01, and 23) exhibiting no dynamic range when the room light is off. Hence, we remove these three illumination variants in our experiments.

4.7.3 Baseline Algorithms

One of the naive ways to apply the relighted images is to perform image sets comparison using the Frobenius norm on the difference between two image sets described as follows:

$$k^* = \underset{i \neq j}{\operatorname{argmin}} \parallel Z_i - Z_j \parallel_F \tag{4.39}$$

where k^* is the identity index and Z_k is the set of relighted images (reconstructed illumination basis). This method considers its canonical topology as a Euclidean space and we refer this approach as a **Naive** method.

Another baseline algorithm adopted in our experiments is the novelty filter [52] which is a de facto [7, 45, 121] face recognition algorithm used widely in reconstructed illumination spaces. The novelty filter can be expressed as:

$$k^* = \underset{k \in gallery}{\operatorname{argmin}} \parallel \hat{x} - Y_k Y_k^T \hat{x} \parallel_2 \tag{4.40}$$

where k^* is the identity index, Y_k is the orthonormal reconstructed basis, and $\hat{x}$ is a novel image.

The other two baseline algorithms are the Canonical Graßmann Horizontal Projection (CGHP) given in Equation (4.38) and the geodesic method [67]. Both of these methods are for image set matching on a Graßmann manifold but consider different aspects of geometry. The CGHP measures the length of tangent directions on the tangent plane of a Graßmann manifold whereas the geodesic method projects a set of images on a Graßmann manifold and computes a geodesic distance.

4.7.4 Results and Findings

Using the illumination model described in Section 4.3, every probe and gallery image is relighted to create nine illuminated images. The recognition results are obtained using the leave-one-out protocol in which one image is considered as a probe image and the rest of the images are treated as gallery images. Results are reported for the portions of the data sets generally considered to be harder. Specifically, the room lights off of CMU-PIE and subsets 3, 4 and 5 of the YaleB and Extended-YaleB data sets. Note that the CSQ, Geodesic, CGHP and Naive algorithms match sets of images whereas the *Novelty* algorithm matches single images.

Method	P1	P2	P3	P4
CSQ	100%	100%	100%	100%
Geodesic	100%	100%	100%	100%
CGHP	100%	100%	100%	100%
Naive	100%	99.44%	99.63%	99.63%
Novelty	99.91%	99.16%	99.63%	99.72%

Table 4.1: Rank One Recognition: Train on the CMU-PIE and test on four CMU-PIE partitions. (Protocol I, Section 4.7.2)

Method	CMU-PIE*	CMU-PIE$^+$
CSQ	100%	100%
Geodesic	100%	99.93%
CGHP	100%	100%
Naive	99.58%	99.65%
Novelty	99.79%	15.97%

Table 4.2: Rank One Recognition: Train on the Extended-YaleB and test on CMU-PIE where * indicates using subset 3 and subset 4 for training, and + indicates using subset 3, subset 4, and subset 5 for training. (Protocol II, Section 4.7.2)

Method	Subset 3	Subset 4	Subset 5
CSQ	99.17%	97.86%	85.79%
Geodesic	95.00%	84.29%	74.21%
CGHP	100%	92.86%	78.95%
Naive	81.67%	62.86%	60.00%
Novelty	53.33%	9.29%	10.53%

Table 4.3: Rank One Recognition: Train on the Extended-YaleB and test on the YaleB. (Protocol III, Section 4.7.2)

Method	Subset 3	Subset 4	Subset 5
CSQ	99.78%	97.88%	51.78%
Geodesic	78.60%	63.71%	29.30%
CGHP	54.28%	32.63%	15.65%
Naive	62.62%	50.00%	33.14%
Novelty	9.23%	5.79%	11.67%

Table 4.4: Rank One Recognition: Train on the CMU-PIE and test on the Combined-YaleB. (Protocol IV, Section 4.7.2)

Tables 4.1, 4.2, 4.3 and 4.4 present results for the experiments defined in Section 4.7.2. Table 4.1 shows all methods perform well when trained and tested on CMU-PIE. The other cases are more interesting and lead us to the following observations.

First, the results from Table 4.2 reveal that a novelty filter is sensitive to the selection of illumination bases while other methods are robust and perform well when trained on portions of the YaleB data and tested on the CMU-PIE data set. A similar finding can be obtained when we train on the Extended-YaleB and test on YaleB data set.

Second, the CGHP algorithm outperforms the novelty filter. This is interesting because CGHP can be considered a high dimensional extension of a novelty filter, using the whole set of a reconstructed basis to perform recognition whereas the novelty filter applies the projection of a novel image to the space spanned by the reconstructed basis. This finding gives an indication that a set-to-set matching is superior to a single image matching.

Third and surprisingly, the naive method does not perform that poorly. The experimental results given in Table 4.2, Table 4.3, and Table 4.4 reveal that the naive method actually outperforms the novelty filter which is considered as a standard way [7, 45, 121] of using the reconstructed illumination basis. While the naive method treats the topology as a Euclidean space, it is still an image set matching method. This finding further supports that image set matching may carry more discriminative information for classification.

Fourth, the observations from Table 4.3 and Table 4.4 reveal that the performance between CGHP and the geodesic method depends on the training set. The CGHP outperforms the geodesic method when similar illumination conditions reside on the training set, otherwise, the geodesic method is more resilient than the CGHP.

Fifth, results in Tables 4.2 and 4.4 suggest an interesting asymmetry when the illumination model is built from one data collection and then performance is tested on the other. Table 4.2 shows that a lighting model trained on YaleB gives rise to perfect results when testing on CMU-PIE for CSQ. Table 4.4 shows that a lighting model trained on CMU-PIE and then tested on YaleB subsets yields CSQ results between 51.78% and 99.78%. We speculate that there is a greater variety of illumination conditions presented in the YaleB data set which gives rise to a

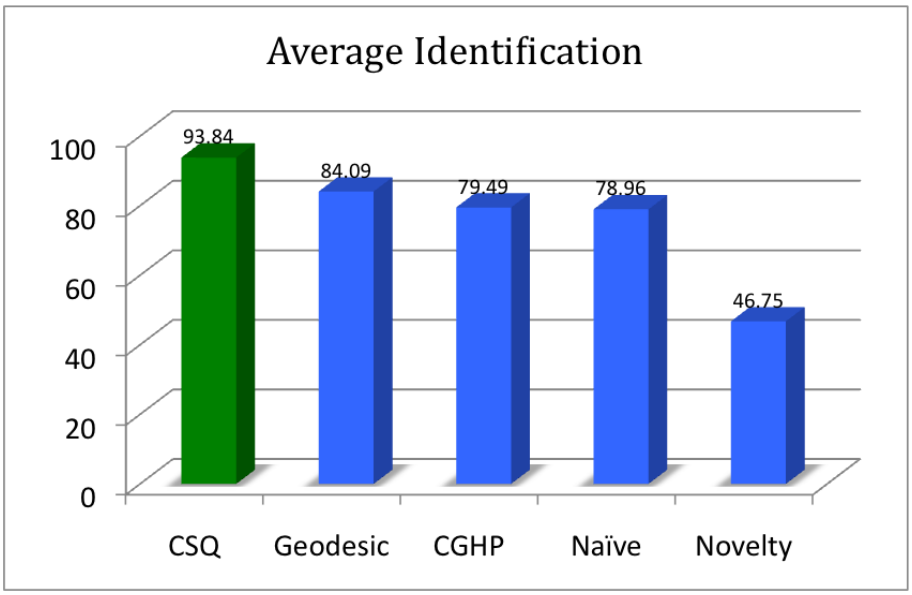


Figure 4.7: Overall performance: average rank one recognition

more powerful lighting model and finally better recognition performance when YaleB imagery is used to construct the lighting model.

It should be noted that the images of subset 5 in YaleB and Extended-YaleB are extremely severe. Most algorithms [38, 45, 59, 121] tested on YaleB and Extended-YaleB do not test this subset. Although our method does not achieve satisfactory results on the subset 5 when we train on the CMU-PIE data set, it is still the best algorithm tested in this work.

To sum up the overall performance, we average all the experimental results from Table 4.1, Table 4.2, Table 4.3, and Table 4.4. The averaged results given in Figure 4.7 provide an indication of the overall performance. As Figure 4.7 shows, CSQ is the top performing algorithm followed by the geodesic method. The reason is related to the characteristic of the chosen manifold and how we set up the illumination variants.

While the order of relighted images does not change the geodesic distance on Graßmann manifolds, it plays a significant role on Stiefel manifolds. The reason why CSQ performs better is that the order of the relighted images matters on Stiefel manifolds. Since we relight an image to a set of fixed order illumination variants, CSQ takes advantages of this fact and outperforms the geodesic distance method. To illustrate the importance of ordering, one more experiment

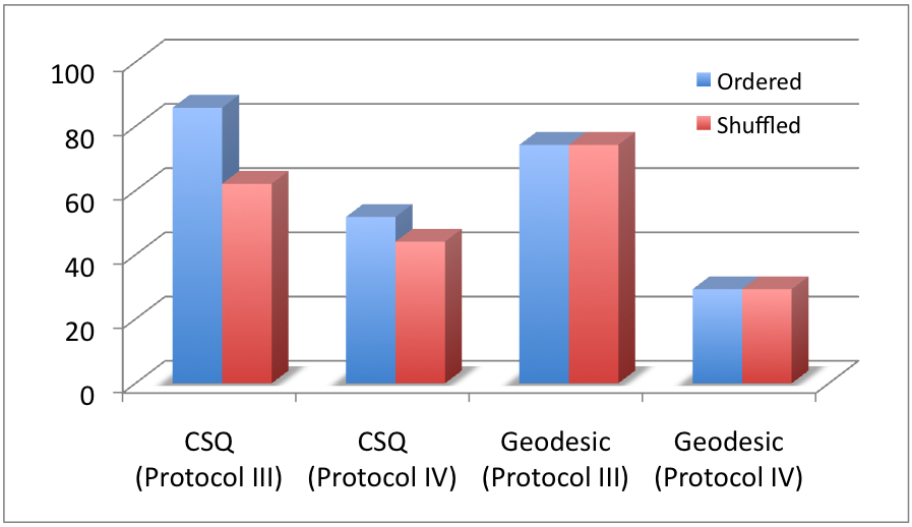


Figure 4.8: The ordering effect on our CSQ and the Geodesic method

was conducted on protocol III and protocol IV using the subset 5 data set where lighting bases were randomly permuted. The results are show in Figure 4.8, and reveal that the recognition rate decreases when we randomly order the relighted images. This observation suggests that whenever the advantage of ordering is presented, CSQ is the choice of representation.

Finally, the proposed canonical Stiefel quotient (CSQ) not only outperforms all other methods tested here, but it is robust to the choice of training sets as well. This finding emphasizes the importance of techniques that account for the underlying geometry of illumination spaces and illustrate that the proper utilization of geometry results in superior face recognition performance.

Chapter 5

Graßmann Registration Manifolds

5.1 Introduction

Registration, registration, and registration [46] has been quoted as being the three greatest challenges in face recognition. Making face recognition algorithms less sensitive to registration errors has received considerable attention [123]. With so much effort paid to registration as a problem, comparatively little work has investigated the structure of face patterns under small registration perturbations. A collection of these perturbed images exhibits an underlying geometric structure which is highly nonlinear and may be discriminative.

In general, face images reside in an abstract image space called a manifold. The nature of the manifold depends fundamentally upon what is assumed to be varying. Small perturbations in registration give rise to high dimensional data points, expressions of the original images, that lie on what is best described as a registration manifold. We will show how a registration manifold may serve as the basis for face recognition and offer levels of performance comparable to the best, often highly trained, algorithms.

Recently, many manifold related algorithms have been proposed. There are two main schools of thought for making use of manifolds. The first is to unfold a manifold whereas the other is to model the manifold directly. Many manifold learning techniques like ISOMAP [104] and LLE [89] attempt to unfold the curved manifold onto a flat surface. Although these manifold learning techniques can be effective in learning the intrinsic structure of a manifold, they require a large amount of training data and dense sampling on the manifold. Such rich training data may

not be available in some real-world applications. It is also not entirely clear how to apply these manifold learning techniques to the task of comparing pairs of still face images.

Another school of thought is to model image manifolds directly. For example, prior work has used image perturbation to synthesize a set of registration images starting from a single image and then used these samples to support modeling of the underlying nonlinear image manifolds [96] [31] [77] [5]. A key distinction between our approach and these methods is that previous methods only seek a single point from a registration manifold for recognition and disregard the underlying geometric structure.

To be more precise, an image with n pixels typically resides in $\mathbb{R}^n$ and distance is measured between pairs of elements in $\mathbb{R}^n$. So, for example, when using the tangent distance [96] or the joint manifold distance [31], the elements may locally shift on the manifold, but ultimately comparison is still between pairs of images drawn from $\mathbb{R}^n$. In contrast, the approach developed in this chapter compares local tangent planes which reside in $\mathbb{R}^{n\times p}$ where p is the number of samples used to characterize the local geometry of the registration manifold.

This distinction is critical in at least two ways. First, we perform recognition in a high dimensional $\mathbb{R}^{n\times p}$ space rather than in $\mathbb{R}^n$. Second, the underlying topological assumptions are different. The earlier work uses an L_2 norm which regards the canonical topology as Euclidean. Our work embeds the entire tangent plane on a Graßmann manifold and uses the geodesic distance accordingly. Therefore, the underlying geometry of the registration manifold is exercised. These distinctions set us apart from previous methods and provide remarkable discriminative power for face recognition.

Since the samples on the registration manifold are derived from a single image, our algorithm works on problems where only a single image per person is available in the probe and gallery sets. In general, recognition given only a single image per person is still challenging [102]. To assess the performance of our method, we compare our approach to eight well-known and recent algorithms using the FERET protocol [83]. The results suggest our algorithm is competitive with the best, achieving a rank one identification of 84.6% on the $Dup2$ probe set. There are two algorithms performing better than our method at 85% and 88.9% on the $Dup2$ data set. However,

unlike our algorithm, both of these algorithms require training. While of course training may be valuable, it also introduces a set of concerns about generalization that non-trained methods avoid.

Our peak performance is achieved using a set of local image features in a fashion similar to that employed by many state-of-the-art algorithms. What happens when the local features are not used and only a holistic representation is applied suggests just how much useful information is captured by the local geometry of the registration manifold. We are not aware of any prior work using a holistic representation that has done better than ours on either $Dup1$ or $Dup2$. We also compare our approach to that of using tangent distances [96] as already mentioned above and our results support further our claim that using the underlying geometry of the registration manifolds yields advantages when performing face recognition.

5.2 Graßmann Registration Manifolds

Image perturbation has been explored and proven to be useful in pattern recognition. What makes our proposed method different from other algorithms is our use of image perturbation. While traditional approaches regard misalignment as noise and correct it through perturbing the original image, our method exploits a set of perturbed images and their underlying geometry such that the representation of an image translates to a tangent space. In this section, we discuss how a registration manifold is formed and embedded on a Grassmann manifold. First, we illustrate the key assumptions for modeling a registration manifold.

5.2.1 Assumptions

The construction of our Graßmann registration manifolds is based upon two assumptions. First, the image space is globally curved but locally Euclidean. This means that the topology of the registration manifold is generally a curved space but locally looks like $\mathbb{R}^n$. More precisely, a point in this space has a neighborhood which is homemorphic to an open set. This space is characterized as locally Euclidean. This assumption allows us to form a tangent space from a registration manifold which exhibits a vector space structure.

The open set property leads to the second assumption that a registration image set is open.

This property implies that every point in the space is an interior point, and that a point can be moved in any direction by a small amount and remains inside the registration image set. This assumption allows us to sample any image on a registration manifold.

5.2.2 Registration Manifolds Formation

Sampling and characterizing a registration manifold are the key steps in our proposed approach. Given a pair of eye coordinates, we determine a set of affine parameters for geometric normalization. The affine transformation maps the (x, y) coordinate from a source image to the (u, v) coordinate of a normalized image. The transformation can be written as follows:

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & dx \\ \sin(\theta) & \cos(\theta) & dy \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & q & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1+s_x & 0 & 0 \\ 0 & 1+s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{5.1}$$

where the first matrix describes rotation and translation, the second matrix represents skew, and the third matrix denotes scaling. These transformed coordinates can be rewritten more compactly as:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} p_1 & p_3 & p_5 \\ p_2 & p_4 & p_6 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{5.2}$$

Equation (5.2) reveals that there are six control parameters for the affine transformation. In this chapter, a set of registration images are sampled by perturbing these six affine parameters as shown in Equation (5.3).

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} p_1+\Delta p_1 & p_3+\Delta p_3 & p_5+\Delta p_5 \\ p_2+\Delta p_2 & p_4+\Delta p_4 & p_6+\Delta p_6 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{5.3}$$

Specifically, we perturb the initial affine parameters with Δp in a $\pm$ range, such that we synthesize 3^6 (729) perturbed images. These 729 images reside on an affine registration manifold $\mathcal{M}$. In our experiments, we employ bilinear interpolation for sampling the registration manifold, and set Δp_1, Δp_2, Δp_3, and Δp_4 as {-0.03, 0, 0.03}, and Δp_5 and Δp_6 as {-3, 0, 3}. From a geometric point of view, we sample the registration images onto a registration manifold, and this registration manifold has a topology associated with it as illustrated in Fig. 5.1.

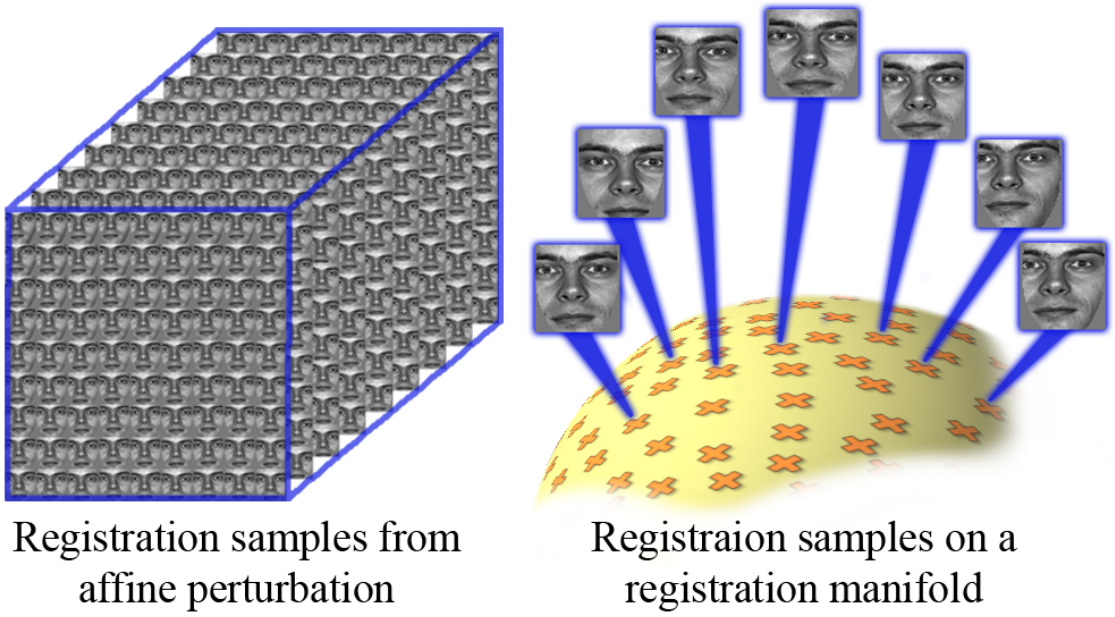


Figure 5.1: Illustration of registration manifold sampling. The cube on the left illustrates the $3^6 = 9^3 = 729$ distinct registration samples and the picture on the right illustrates how samples are then arrayed upon a curved registration manifold.

5.2.3 Registration Manifolds Formation

The affine registration manifold has a nonlinear structure [96, 92]. One way to utilize the sampled registration manifold is to assume local linearity and explore its tangent space. This is essentially a local linear approximation of a hypersurface. By the mean value theorem [97], we can approximate a tangent vector using a nearby secant. A collection of these approximated tangent vectors at a point $x \in \mathbb{R}^n$ forms a tangent space.

A tangent plane could be centered on any interior point on a registration manifold, but given our problem formulation, the logical choice for the base point x is the canonical sampled image ($\Delta p_i = 0$ in Equation (5.3)). The selected base point x is assumed to be an interior point that has a neighborhood homeomorphic to an open ball. Then, given k nearest neighbors $\{x_1^*, x_2^*, \ldots, x_k^*\}$ where $x_i^* \in \mathcal{M}$, a tangent space centered at x on a registration manifold $\mathcal{M}$ may be represented as:

$$T_x\mathcal{M}: \quad x + \text{span}\{x - x_1^*, x - x_2^*, \ldots, x - x_k^*\} \tag{5.4}$$

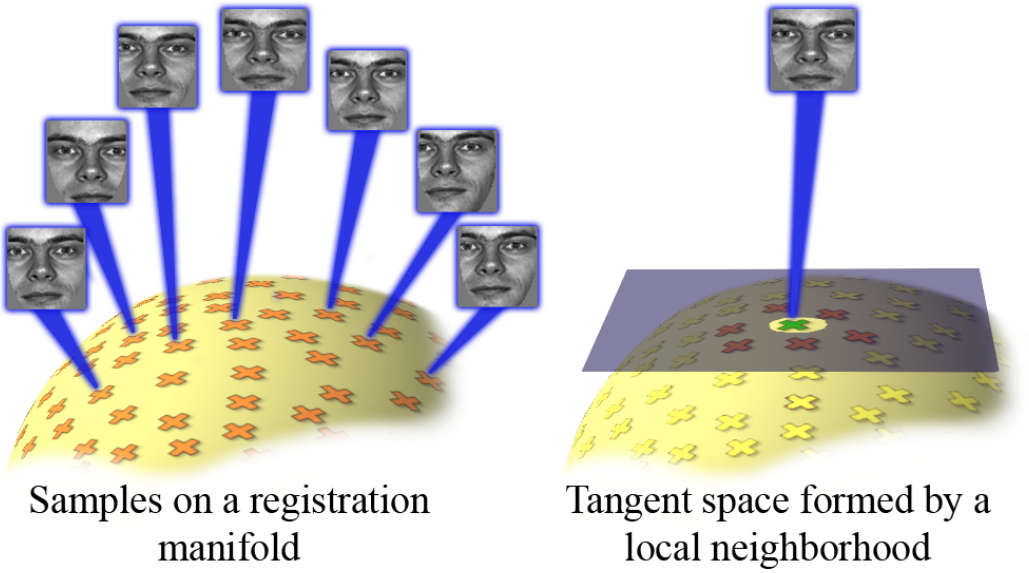


Figure 5.2: The use of local Euclidean distance to form a tangent space. A tangent space of a registration manifold is formed using a local neighborhood centered around the canonical image. In the illustration, six neighbors are shown in orange surrounding the canonical image which is indicated by the green cross. It is important to note that this figure is simplified for the sake of illustration, and in practice we take hundreds of samples on the registration manifold and what is here shown as a plane is in practice a linear subspace of many dimensions - upwards of 100 in most cases.

where $\{x - x_1^*, x - x_2^*, \ldots, x - x_k^*\}$ are the approximated tangent vectors around x. These approximated tangent vectors are the direction vectors from the affine transformations. In addition, the tangent space $T_x\mathcal{M}$ has a vector space structure and any point on this tangent space can be reconstructed as:

$$x + \sum_{i=1}^{k} \alpha_i (x - x_i^*) \tag{5.5}$$

Generally, a manifold is globally curved but locally Euclidean. We use this property to guide our choice of k nearest neighbors. Specifically, we assume local linearity about x in order to select the k nearest neighbors using Euclidean distance:

$$x_k^* = x_{k-1}^* \bigcup \underset{x_j \notin x_{k-1}^*}{\operatorname{argmin}} \parallel x - x_j \parallel_2^2 \tag{5.6}$$

where $x_0^* = \{\}$ and x_k^* is the set of k nearest neighbors. Thus, a tangent space is a set from a local neighborhood centered around x of a registration manifold as depicted in Figure 5.2.

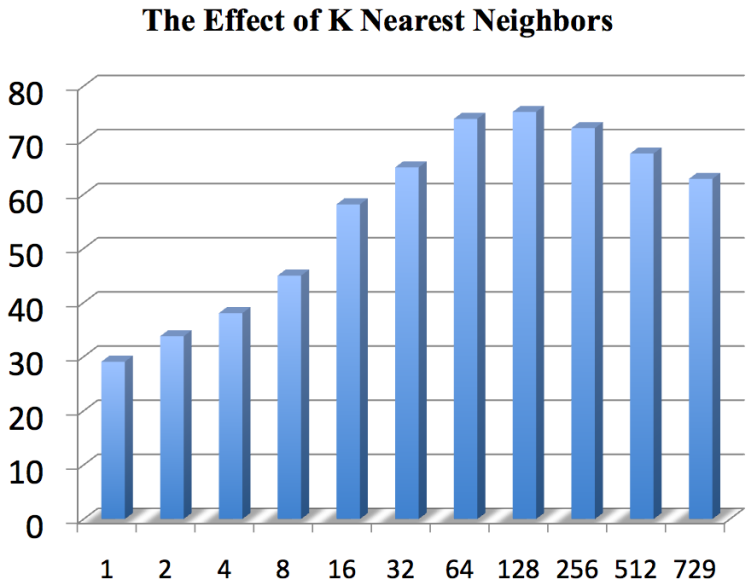


Figure 5.3: The effect of varying the number of nearest neighbors used to define the tangent plane. The horizontal axis is laid out by categories, one for each value for k tested. The vertical axis is the rank one identification rate for the FERET $Dup2$ probe set.

Although we apply the same image perturbation to generate each registration image, the k nearest neighbors for different face images are usually different. This is because every face image has distinct facial structures so that transforming the face image may change the relative Euclidean distances between different perturbed images. This process allows us to capture the local structure of a registration manifold and makes the local neighborhood adaptive. In early experiments, we used a fixed set of registration images always derived from the same alignment parameters and hence did not capture this person-by-person local variation. When the associated tangent spaces were used for recognition, the results were inferior to those presented here.

Selecting the number of nearest neighbors to approximate the tangent space has significant implications. At the extreme low end, $k = 1$, minimizing the chordal distance equates to maximizing the correlation between pairs of images (Equation (3.43)). At the other extreme, taking hundreds or thousands of samples is not only computationally burdensome; at some point the local linearity assumption is stretched beyond the breaking point, and the discriminative power starts to decrease. The effect of different choices for k on the FERET $Dup2$ data set is shown in Figure 5.3. As Figure 5.3 depicts, the rank 1 identification starts to decrease when k is larger

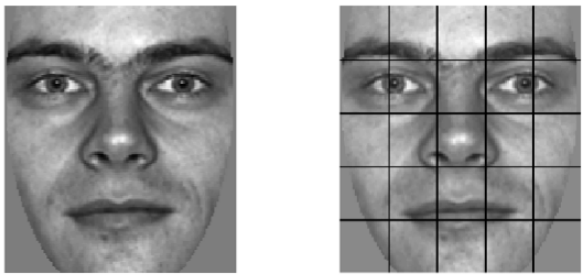


Figure 5.4: Examples of a holistic face and local regions

than 128. In our experiments, we choose $k = 100$. The value of k was initially set to 81 which came from our initial eye perturbation experiments (3^4 = 81). Subsequently, we changed the value of k to 100. As Figure 5.3 suggests, any value around 100 can be expected to perform well. It should be noted that the value of k is related to how curved a registration manifold is and how we sample the registration manifold.

To review what we have developed so far, recall that a Graßmann manifold $\mathbb{G}_{n,p}$ is a set of p-dimensional linear subspaces of $\mathbb{R}^n$. Because a tangent space admits a vector space structure, and the approximated tangent vectors are the bases spanning the subspace, we use these tangent vectors to embed a linear subspace on a Graßmann manifold. Thus, our proposed method performs recognition in $\mathbb{R}^{n\times p}$ dimensions while most existing algorithms performs recognition in $\mathbb{R}^n$. The benefits of embedding a tangent space on a Graßmann manifold are that the properties and distance metrics are well studied. Since we embed the registration images on a Graßmann manifold, we call the resulting manifold the **Graßmann Registration Manifold (GRM)**.

5.3 Image Features and Image Preprocessing

A holistic image is regarded as the whole face represented by a single high dimensional vector whereas local regions are derived from regular sampling patterns, for example a grid laid over the face. In this chapter, we employ a holistic image and local regions for face recognition. Local

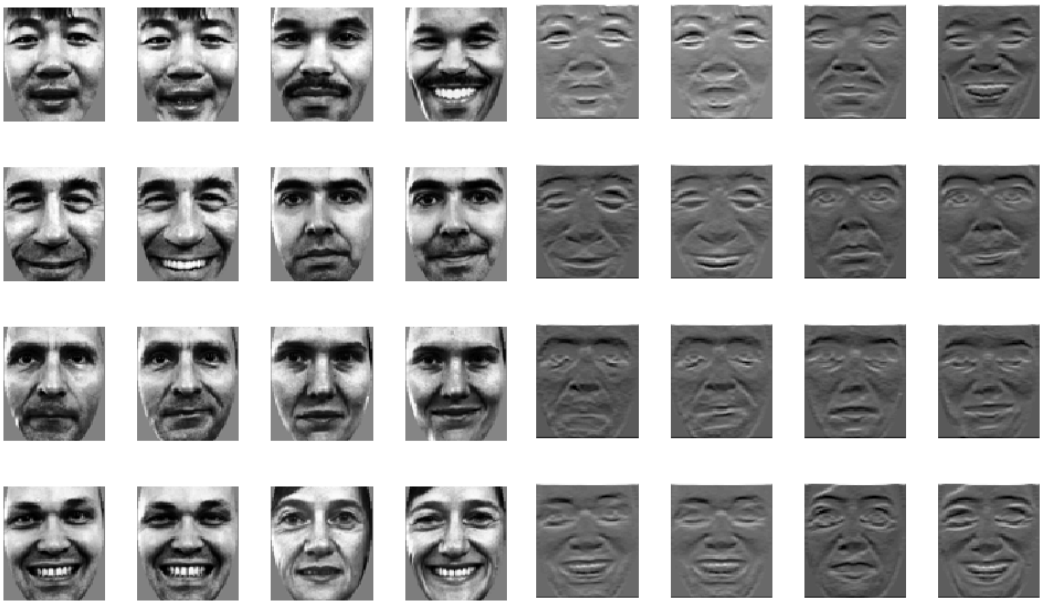


Figure 5.5: Examples of original face chips and Gabor processed face chips

regions are extracted from a 5 $\times$ 5 facial window.[1] Examples of a holistic face and local regions are given in Figure 5.4.

For each probe and gallery image, the 729 registration variants are sampled using eye coordinates provided with the FERET data and using the perturbation process described in Section 5.2.3. In addition, an elliptical mask is applied to remove the background. Histogram equalization and a Gabor filter [55] are then applied to the holistic image. To create a greater degree of independence, the local regions are used without histogram equalization or the application of the Gabor filter.

Gabor filters have a long history of use in face recognition, and while most algorithms [93, 103, 124] apply filters tuned to multiple scales and orientations, we have found using a single filter sensitive to horizontal features on the face yields very good results. We think this makes sense because much of the facial structure that is useful for identification is dominated by

[1]Since most of the lower left and lower right regions are covered by a mask in the 5 $\times$ 5 facial window, we eliminate these two regions from our feature set.

Algorithm 1 : Image Matching using Graßmann Registration Manifolds

1: Determine initial registration parameters $p_1, \ldots, p_6$ for each image using the eye coordinates. (Equation (5.2))
2: Sample the affine registration manifold by perturbing the affine parameters. (Equation (5.3))
3: Find the k nearest neighbors $\{x_1^*, x_2^*, \ldots, x_k^*\}$ from the registration manifold. (Equation (6.11))
4: (Optional) Apply histogram equalization and a Gabor filter. (Equation (5.7))
5: Construct the tangent space. (Equation (5.4))
6: Embed the approximated tangent space and compute canonical angles. (Equation (3.44))
7: Compute the chordal distance. (Equation (3.48))

horizontal components.

In general, a DC-free Gabor filter is defined as follows:

$$\psi_{u,v}(z) = \frac{\| k_{u,v} \|^2}{\sigma^2} e^{(-\|k_{u,v}\|^2 \|z\|^2 / 2\sigma^2)} \left[e^{ik_{u,v}z} - e^{-\sigma^2/2} \right], \quad k_{u,v} = \frac{k_c}{f^v} e^{i\phi_u} \tag{5.7}$$

where u and v are the control parameters for orientations and scales, respectively, and z is the position. In our experiments, we set the $f = \sqrt{2}$, $k_c = \frac{4\pi}{5}$, $\sigma = \frac{3\pi}{2}$, $v = 0$, and $\phi_u = 0$. Examples of Gabor processed face chips are given in Figure 5.5.

5.4 The Graßmann Registration Manifold Algorithm

Our face recognition algorithm includes seven major elements enumerated in Algorithm 1. To summarize, for all probe and gallery images the affine parameters are perturbed to sample the registration manifold. Next the k nearest registration images, neighbors, of the original image are found. Optionally, histogram equalization and a Gabor filter may be applied to these images. Tangent vectors are then computed from the selected k nearest neighbors and a tangent space is formed using the tangent bases. The probe tangent space and the gallery tangent space are projected onto the Graßmann manifold where the distance between these two tangent spaces is computed using the chordal distance.

This embedding process in general and the use of geodesic distance on a Graßmann manifold in particular are depicted in Figure 5.6. To reiterate what we have mentioned before, each element embedded on a Graßmann manifold represents a tangent space in $\mathbb{R}^{n \times p}$ of a registra-

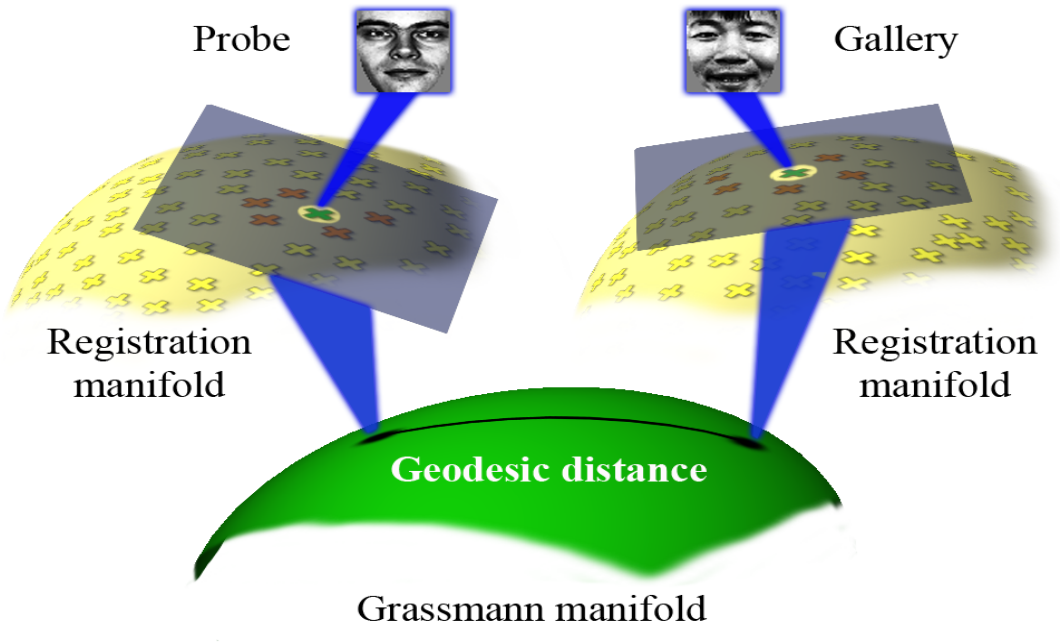


Figure 5.6: The embedding process for the proposed method. Tangent spaces of registration manifolds are embedded on a Graßmann manifold where the geodesic distance, specifically the chordal distance, is computed.

tion manifold. Consequently, recognition is performed in a higher dimensional space rather than using a single image in $\mathbb{R}^n$.

5.5 Many to few Matching Strategy

Initial studies of cumulative match curves indicate that 95% of the correct identifications are included in the top 10% of gallery candidates using just the holistic image matching and $k = 16$, i.e. 16 registration manifold samples. Consequently, considerable time can be saved with our proposed many to few matching strategy. This matching strategy uses a small number of nearest neighbors to form tangent planes and compare a probe image to the entire gallery. The number of nearest neighbors is then increased at the same time that the number of gallery candidates considered is decreased. As a consequence, only a small number of gallery candidates need be compared using tangent planes derived from the full $k = 100$ set of registration variants. Since we apply the many to few matching strategy to a holistic representation, we call this algorithm the **GRM-Holistic** depicted in Algorithm 2.

Algorithm 2 : The GRM-Holistic Algorithm

1: Construct a tangent plane for every probe and gallery using $k = 16$ nearest neighbors.
2: Embed tangent planes on a Graßmann manifold.
3: Compute chordal distances between all probe and all gallery images.
4: **for** each probe image p_i **do**
5: Sort the gallery by increasing chordal distance.
6: Retain in G_i^{k16} the 10% of the gallery closest to p_i.
7: Construct tangent planes for p_i and gallery images in G_i^{k16} using $k = 32$ neighbors.
8: Sort G_i^{k16} by increasing chordal distance using $k = 32$ tangent planes.
9: Retain in G_i^{k32} the 5% of the gallery closest to p_i (50% of G_i^{k16}) .
10: Construct tangent planes for p_i and gallery images in G_i^{k32} using $k = 100$ neighbors.
11: Sort G_i^{k32} by increasing chordal distance using $k = 100$ tangent planes.
12: Return in G_i^{k100} the ranked gallery matches to p_i.
13: **end for** // Returns for each probe image a ranked portion of the gallery.

Algorithm 3 : The GRM-Local Algorithm

1: **for** each local feature L_j summarized in Figure 5.4 **do**
2: **for** each probe image p_i **do**
3: Use as the gallery G_{ij} the gallery G_i^{k100} created by Algorithm 2.
4: Construct tangent planes for L_j of p_i and gallery images G_{ij} using $k = 100$.
5: Compute the chordal distances between p_i and gallery images G_{ij}.
6: Return in G_{ij} the ranked gallery matches to p_i of L_j.
7: **end for** // Returns for each probe image a ranked portion of the gallery.
8: **end for** // Return the top rank for a local feature
9: Apply a majority voting scheme using all features.

As Algorithm 2 illustrates, a probe image is initially compared to the entire gallery using tangent spaces derived from just 16 registration samples. The comparisons are carried out by computing chordal distance between tangent planes of 16 dimensions. The 10% of the gallery candidates found to be most similar to the probe in this first pass is then compared using 32 registration variants. Finally, the top 5% gallery candidates are compared using 100 registration samples. Note that each probe image has its own gallery candidates. Because we perform recognition in a $\mathbb{R}^{n\times p}$ dimension, reducing the dimension p allows us to perform a full search efficiently. As the dimension p increases, the number of gallery candidates reduces to only 5%. Consequently, considerable time can be saved. At this point, the holistic algorithm stops and the resulting top ranked gallery candidate is taken as the rank one match for each probe image.

The local feature algorithm is an extension to the GRM-Holistic algorithm that first carries out all the steps already described for the GRM-Holistic algorithm (Algorithm 2). Then, utilizing the gallery candidates found by the GRM-Holistic algorithm, for each local feature described in Figure 5.4, we form the tangent planes using 100 registration samples for a probe and its associated gallery candidates. The chordal distances between these tangent planes are computed and become the basis for comparing the local features in probe and gallery images. We call this matching algorithm **GRM-Local** and it is described in Algorithm 3.

Applying the many to few matching strategy to the holistic image and subsequently to all local features offers a dramatic reduction in run-time. Overall, the proposed matching strategy speeds up the classification process by roughly a factor of 500, (20×25).[2] The final classification is determined by a majority voting scheme [56] using all features.

5.6 Comparative Evaluation to The-State-Of-The-Art

5.6.1 Data Collection

The performance of our proposed algorithm is compared to well-known and recent algorithms on the FERET database [83]. The frontal view imagery of the FERET database is divided into 5 categories: Fa, Fb, Fc, $Dup1$, and $Dup2$, containing 1,196, 1,195, 194, 722, and 234 faces, respectively. Both Fa and Fb are taken in the same day with the same illumination condition but with different facial expressions. Fc is taken at the same day as Fa but with different illumination condition. $Dup1$ is acquired on different days from Fa. $Dup2$ is acquired at least one year apart from Fa. Following the FERET protocol, Fa is always the gallery and Fb, Fc, $Dup1$, and $Dup2$ are used as probe sets.

5.6.2 Prior Art on the FERET Database

The Elastic Bunch Graph Matching (EBGM) algorithm [114] was one of the top algorithms for the FERET database for more than half of a decade until the Local Binary Pattern (LBP) [3]

[2] 5% candidates and 25 features

was introduced. Since then, many LBP-like algorithms have been reported and continue to push the performance envelope for FERET. Zhang et al. [119] applied Gabor filters to extract local patterns and encoded them as phase-quadrant and XOR patterns. Tan and Triggs [103] proposed to combine 40 Gabor responds and LBP features, and projected it on a discriminant space using kernel discriminative common vectors. Shan et al. [93] employed 40 Gabor filters for feature extraction and constructed an ensemble piecewise LDA classifier for each LBP segment. Zou et al. [124] used a large set, $4,172$, of Gabor jets and achieved excellent results.

Liu et al. [65] created 9 images by horizontally and vertically shifting samples (left/right and up/down). The authors view these images as the basis for linear spatial filters because the 9 translated images make up a filter mask. Subsequently, the subspace distance between local patches is computed and the aggregated score is used for final classification. We include this algorithm in our comparison because it involves subspaces defined by different registration samples, and is thus related to our own work. However, Liu et al. only presented their approach in the context of linear spatial filters, whereas we consider the underlying geometric interpretation. As such, we make use of local linearity and do not choose the same image set all the times.

Table 5.1 summarizes the published results of the above algorithms and three variants of our proposed method for the FERET database. The GRM-Holistic uses only a holistic representation, the GRM-Local exploits all local features, and the GRM-Combined employs the holistic representation and all local features. The rightmost column indicates whether an algorithm requires training. As Table 5.1 reveals, many algorithms perform very well on the Fb and Fc data sets where the probe and gallery images are taken in the same day. The performance of our GRM method on these two data sets is about 98%. On the other hand, the $Dup1$ and $Dup2$ probe sets are more challenging even for the most modern algorithms. To better visualize the ranking for $Dup1$ and $Dup2$, the results are shown from worst to best in Figure 5.7.

5.6.3 Results with the Holistic Representation

Using the entire face image, our GRM-Holistic achieves 94.1%, 92.8%, 70.8% and 76.9% rank one identification for Fb, Fc, $Dup1$, and $Dup2$, respectively. These results already outperform half of the top algorithms and are among the best for non-trained algorithms. It should be noted

Table 5.1: Rank 1 identification on the FERET database

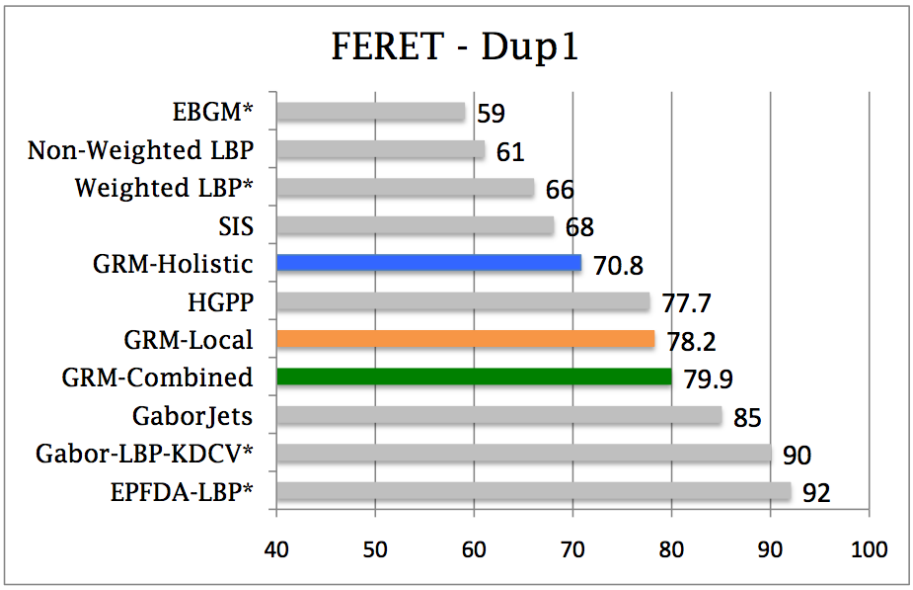
Methods	Fb	Fc	Dup1	Dup2	Trained
EBGM [114]	95.0	82.0	59.0	52.0	Yes
Weighted LBP [3]	97.0	79.0	66.0	64.0	Yes
EPFDA-LBP [93]	99.6	99.0	92.0	88.9	Yes
Gabor-LBP-KDCV [103]	98.0	98.0	90.0	85.0	Yes
Non-Weighted LBP [3]	93.0	51.0	61.0	50.0	No
HGPP [119]	97.6	98.9	77.7	76.1	No
GaborJets [124]	99.5	99.5	85.0	79.5	No
SIS [65]	91.0	90.0	68.0	68.0	No
GRM-Holistic	94.1	92.8	70.8	76.9	No
GRM-Local	97.2	97.4	78.2	80.8	No
GRM-Combined	97.7	97.9	79.9	84.6	No

that our GRM-Holistic is the only holistic method presented in Table 5.1. To our knowledge, these results are the best on the FERET $Dup1$ and $Dup2$ data sets for any holistic representation. This contribution suggests that our method can perform well without specific domain knowledge, thus it is extensible and generic. Additionally, the proposed GRM-Holistic ranks second among all non-trained algorithms for the $Dup2$ data set shown in Figure 5.7b, the probe set for which most algorithms have the greatest difficulty. The only non-trained method performing better than our GRM-Holistic is the GaborJets method [124] which employs a large amount of local features (4,172 Gabor jets).

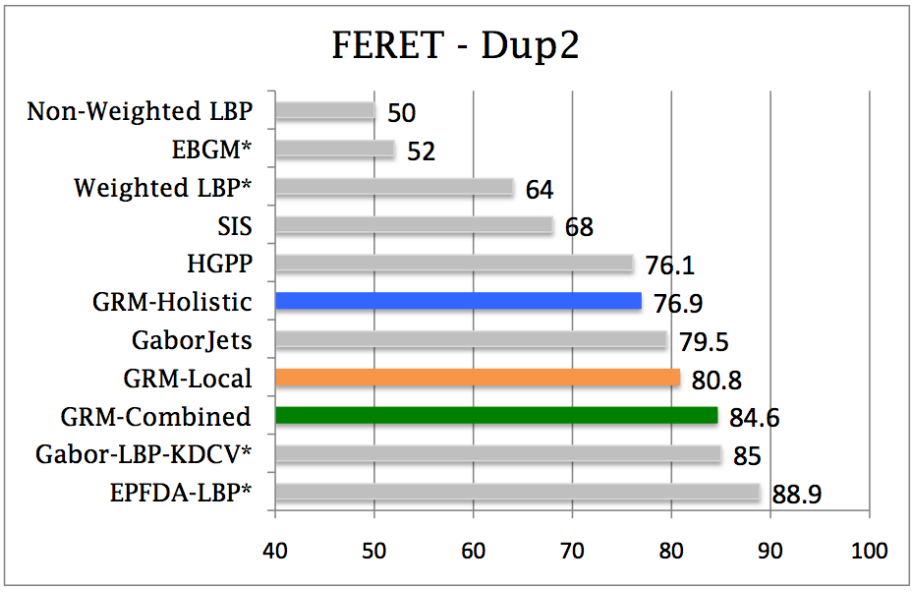
Given the sizes of these probe sets and making the relatively simple assumption that rank one identification success/failure may be modeled as a binomial, it is possible to put confidence intervals on these rank one identification rates [72, 12]. Specifically, the average confidence intervals given the α value as 0.05 for the rank one identification are about $\pm 1\%$, $\pm 4\%$, $\pm 3\%$, and $\pm 5.5\%$ for the Fb, Fc, $Dup1$, and $Dup2$, respectively.

5.6.4 Results with Holistic + Local Representations

There is a general agreement that peak performance is typically achieved using local features. This finding can also be observed from our GRM-Holistic and GRM-Local in Table 5.1 and Figure 5.7. However, the reasons are less consistent. It has been argued that local features



(a)



(b)

Figure 5.7: Rank 1 Identification on the FERET Dup1 (a) and Dup2 (b) data sets for the selected algorithms (Non-Weighted LBP [3], EBGM [114], Weighted LBP [3], SIS [65], HGPP [119], GaborJets [124], Gabor-LBP-KDCV [103], and EPFDA-LBP [93]) where * indicates trained methods

are generally less sensitive than global features to appearance changes [102, 124]. We stress an additional reason why the use of local features can achieve better recognition results in face recognition. Recall that we described our local features in Section 5.3. Figure 5.8 shows the rank one identification achieved when a GRM algorithm is constructed using each individual feature. The associated feature index is given in Figure 5.9.

Neither space allows us to identify each feature, nor is it actually that important. What is important is first to note that the left most column in Figure 5.8 is the holistic image as a feature, and it is consistently one of the best individual features. Hence, we can dismiss any thought that a single local feature consistently does much better than the holistic representation. Second, there is a fair amount of apparently random variation between features and between probe sets.

More specifically, a careful study of Figure 5.8 reveals that no single local feature consistently ranks in the top three for all data sets. Consequently, the strength of using local features comes from the combination of independent decisions boosting recognition performance [56]. A good example is our $Dup1$ results shown in Figure 5.8. As the $Dup1$ results in Figure 5.8 shows, all local features perform worse than the holistic image, nevertheless, combining all these features boosts the recognition result from 70.8% to 79.9% (a green dash line). A similar observation has been reported using weighting subspaces [24]. In this chapter, we use the majority voting rule [56] to combine all the feature outcomes.

Using the whole image plus local features, the GRM-Combined algorithm achieves 97.7%, 97.9%, 79.9% and 84.6% rank one identification for Fb, Fc, $Dup1$, and $Dup2$, respectively. As Figure 5.7a shows, our GRM-Combined obtains 79.9% rank one identification which ranks second among all non-trained methods on the $Dup1$ data set. As the $Dup2$ data set, to our knowledge, the 84.6% rank one identification is the best result for all non-trained algorithms and third among all algorithms depicted in Figure 5.7b.

5.7 Registration Problem Revisited

There are many ways to use registration variants. Most of the traditional techniques [123] apply registration variants to easing registration errors. Considering registration variants as a source

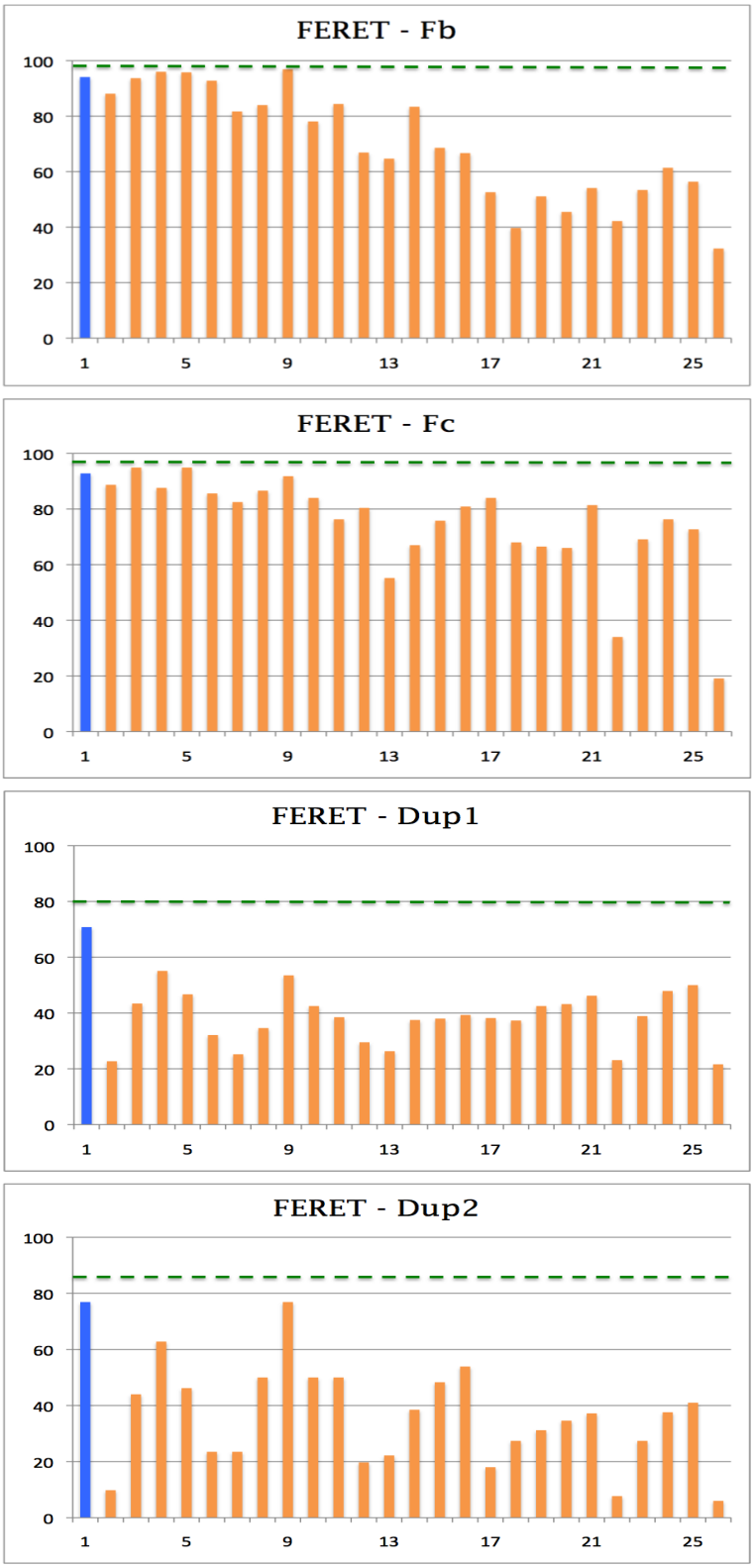


Figure 5.8: Identification results for individual feature (Blue : Holistic, Orange : Local regions, Green : Combined) where the horizontal axis is the feature number and the vertical axis is the rank one identification

Figure 5.9: The indices of GRM features (holistic face and local regions)

for classification may seem odd since they all come from the same image. Why is using the registration variants for recognition better than finding one alignment image? The answer may lie in the distinction between the two paradigms. In this section, we will emphasize the discrepancies between our method and the alignment methods.

There are two families of techniques for image alignment. The first is to search the most probable registration images while the other employs the registration variants to reconstruct an alignment image. First, we briefly describe these alignment algorithms as follows.

5.7.1 Brute Force Approach

Perhaps, the most straight forward approach to overcome an alignment error is to try all plausible registration images. If the algorithm tries enough possibilities, it will stumble upon the correct one at some point. We will call this method a brute force registration algorithm and it will return the best score achieved over a range of typically hundreds of alternative registration variants. Indeed, for the experiments that follow the 729 registration samples already defined will be used.

Formally, the scoring operation for the brute force algorithm can be expressed as:

$$S_{cor}(x, y) = \max_{1 \le i \le q} \max_{1 \le j \le q} \{\mathbf{Corr}(x^{(i)},\ y^{(j)})\} \tag{5.8}$$

where x, $y \in \mathbb{R}^n$ are the probe and gallery images, **Corr** denotes the normalized correlation[3], q

[3]We use **Corr** because it is a de facto standard. Other similarities or distance measures can also be applied.

is the number of registration images (729 in our experiments), and i and j are the indices for the registration variants.

As Equation (5.8) shows, each probe and gallery pair has $q \times q$ matching scores: $531,441$ in our experiments. Undoubtedly, the brute force approach is computationally expensive, and our motivation for considering it rests in what it can tell us about registration as a source of errors. In other words, if we can match a large amount of registration variants, how much can an algorithm improve?

5.7.2 Tangent Distances

Another family of techniques for the registration problem is to reconstruct an alignment image using a set of registration variants. One such method is the well-known tangent distances [96]. Tangent distances have found a number of successful applications including handwritten digit recognition [96] and face recognition [108]. While both tangent distances and our GRM make use of tangent spaces, there are some fundamental differences. These differences yield significant improvement on face recognition performance. First, let us briefly review the key ideas of tangent distances. Given a gallery image $x \in \mathbb{R}^n$ and a probe image $y \in \mathbb{R}^n$, and their associated tangent vectors, tangent distances can be computed as one sided tangent distance as well as two sided tangent distance. We review these two methods in the following subsections.

5.7.2.1 One Sided Tangent Distance

The one sided tangent distance employs the first order Taylor expansion to approximate a tangent plane at a gallery image. Every element of this tangent plane is a linear combination of tangent vectors. The key idea of the one sided tangent distance is to seek a point from this tangent plane such that the Euclidean distance from this point to a probe image is minimized. Formally, the one sided tangent distance can be formulated as follows:

$$TD_1(x, y) = \min_{\alpha} \parallel x + T_x\alpha - y \parallel_2^2 \tag{5.9}$$

where T_x is the matrix containing tangent vectors of x, and α is the weighted coefficient. As Equation (5.9) shows, the one sided tangent distance computes a Euclidean distance between

a point from a tangent plane ($x + T_x\alpha$) and an image y. The coefficient α is determined by minimizing this Euclidean distance. The illustration of the one sided tangent distance is given in Figure 5.10.

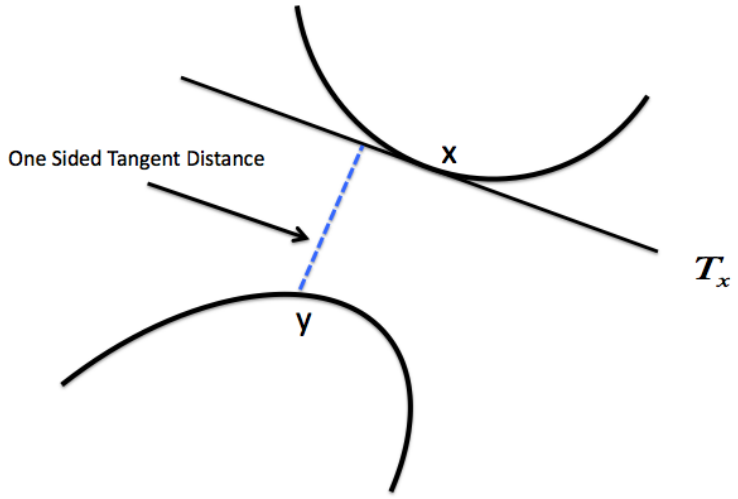


Figure 5.10: Illustrations of the one sided tangent distance : Euclidean distance from a point to a tangent plane

5.7.2.2 Two Sided Tangent Distance

While the one sided tangent distance utilizes a tangent plane typically derived from a gallery image, the two sided tangent distance makes use of tangent planes from both a gallery image and a probe image. In other words, the two sided tangent distance forms two tangent planes and seeks points from these tangent planes such that the Euclidean distance between a point of a probe tangent plane and a point of a gallery tangent plane is minimized. Figure 5.11 depicts the use of the two sided tangent distance. Mathematically, the two sided tangent distance can be defined as follows:

$$TD_2(x, y) = \min_{\alpha,\beta} \parallel x + T_x\alpha - y - T_y\beta \parallel_2^2 \tag{5.10}$$

where T_x and T_y are the matrices containing tangent vectors of x and y, and α and β are the associated weighted coefficients, respectively. As Equation (5.10) reveals, the two sided tangent distance computes a minimum Euclidean distance between two points from two tangent planes.

Since the two sided tangent distance exercises two tangent planes, it has some similarity to our GRM. Next, we stress the distinction between our GRM and the two sided tangent distance.

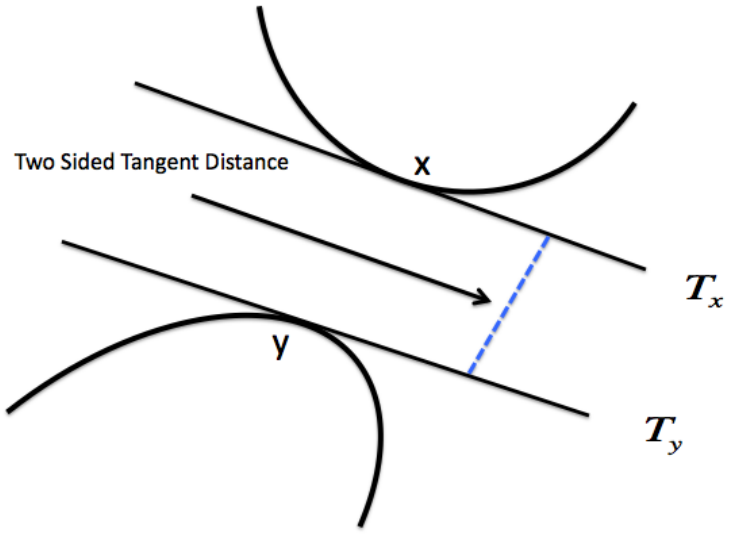


Figure 5.11: Illustrations of the two sided tangent distance : Euclidean distance from a point of a tangent plane to a point of another tangent plane

5.7.3 Brute Force, Tangent Distances, and Graßmann Registration Manifolds

The commonality between the brute force approach, tangent distances, and our GRM is that they all employ registration variants. The differences between these methods are summarized as follows:

- Both the brute force approach and tangent distances perform recognition in $\mathbb{R}^{n\times 1}$ while our GRM performs recognition in $\mathbb{R}^{n\times p}$ (p is the number of tangent vectors). Therefore, our GRM performs recognition in a high dimensional space.
- Both the brute force approach[4] and tangent distances perform recognition using a Euclidean distance[5] while we consider local linearity, project the tangent planes onto a Graß-

[4]The normalized correlation is related to a Euclidean distance.

[5]Because images reside on a manifold, only nearby images exhibit a Euclidean structure.

Table 5.2: GRM vs Brute Force Approach and Tangent Distances (FERET database)

Methods	Fb	Fc	Dup1	Dup2
Brute Force Approach (Holistic)	69.0	60.3	55.0	54.7
One Sided Tangent Distance (Holistic) [96]	68.2	33.0	39.2	33.3
Two Sided Tangent Distance (Holistic) [96]	81.3	18.0	44.5	41.9
GRM-Holistic	94.1	92.8	70.8	76.9

mann manifold, and compute a geodesic distance. Hence, the underlying geometry is exercised.

- While both the two sided tangent distance and our GRM employ two tangent planes, the two sided distance considers a point on a tangent plane and computes a Euclidean distance between points on two tangent planes whereas our GRM utilizes the entire tangent plane and computes a geodesic distance between two tangent planes.

- Both the brute force approach and tangent distances apply the registration variants to correct the image alignment while our GRM uses registration variants to form a feature representation (tangent space).

To show the advantages of our GRM over the brute force approach and tangent distances, we assess the face recognition performance using the holistic image on the FERET database. To have consistent comparisons, all images are preprocessed by our Gabor filter discussed in Section 5.3. We employ 729 registration variants for the brute force approach and the same 100 tangent vectors as our GRM method to form a tangent space for the tangent distances.

The rank one identification results are summarized in Table 5.2. Our proposed GRM method outperforms the brute force approach as well as the one sided and two sided tangent distances. These experiments reveal the importance of using the underlying geometry on manifolds that could result in significant enhancement in face recognition performance. While methods aimed at overcoming registration errors are important, further study of registration variants and their underlying geometry may provide us with new insights for how characterizing image variability and construct more effective recognition algorithms.

Chapter 6

Graßmann Product Manifolds

6.1 Introduction

Human-computer interaction has attracted great attention in recent years [75, 106, 84] due in part to many potential applications. Action classification is one key aspect of human-computer interaction, and a variety of methods have been proposed to construct action classifiers. Bissacco et al. [14] employed an ARMA model for human gait recognition. Schüldt et al. [91] combined local features and SVMs for human action classification. Turaga et al. [107] applied Procrustes distances on special manifolds for activity recognition. Recently, Laptev et al. [57] proposed a method using spatio-temporal bag-of-features combined with multi-channel SVMs for action classification. Despite these efforts, reliable action classification remains a hard problem because of the complexity of human motions. To address this concern, more powerful tools are needed, and one such tool is multilinear algebra.

Multilinear algebra is a mathematical framework for high order tensors which capture multiple factor variations and interactions. Tensor computing has been successfully applied to many computer vision applications such as face recognition [109, 42, 33, 88], visual tracking [63], and action classification [48]. However, the advantages of representing a tensor on a product manifold in the context of classification have not been explored. In this chapter, we represent a video as a 3rd order tensor and demonstrate that the geometric structure of the tensor space is discriminative for action classification.

It is known that multidimensional data can be considered as a high order tensor. Previous

methods [109, 42, 33, 88, 48] often learn a projection to characterize a lower dimensional tensor subspace and apply discriminant analysis. Such techniques are usually complicated due to the nature of the learning algorithms. In addition, they require a large amount of training data and may suffer from generalization problems. With so much effort paid to the learning algorithms, comparatively little work has investigated the underlying geometry of the tensor space.

The method proposed in this chapter employs tensors to perform action classification from a different perspective. First, it is a non-trained method, and so avoids training data and generalization problems. Second, we focus attention on the geometric structure of the tensor space, and this in turn provides insight into the existing multilinear algebra, its underlying geometric interpretation, and how this geometry provides a robust basis for classification.

Our approach begins by abstracting an N order tensor as a point on a product manifold where the number of factors is given by the order of the tensor. Because each factor of the tensor can be characterized as an orthogonal matrix via a decomposition procedure, it is represented on a Graßmann manifold. However, traditional Higher Order Singular Value Decomposition (HOSVD) [27] does not factorize a space which preserves the geodesic distance in the context of video classification. It is therefore helpful to modify the common definition of HOSVD, and with the modified HOSVD, each factor manifold is related to a single order of the tensor spanned by the column space. As such, an N order tensor yields N factor manifolds.

The proposed approach draws upon the fact that the geodesic on a product manifold is equivalent to the Cartesian product of geodesics from multiple factor manifolds. In other words, elements of a product manifold are from the set of all elements on factor manifolds. Action classification is then performed on the basis of geodesic distance on a product manifold associated with an action video.

The most important contribution of this chapter is the presentation of a new way of relating data tensors and the product manifold geometry to the practical task of action video classification. Using the geodesic distance on a product manifold to compare videos, we show that a simple nearest neighbor classifier can perform very well: a match for the best highly trained algorithms in the literature.

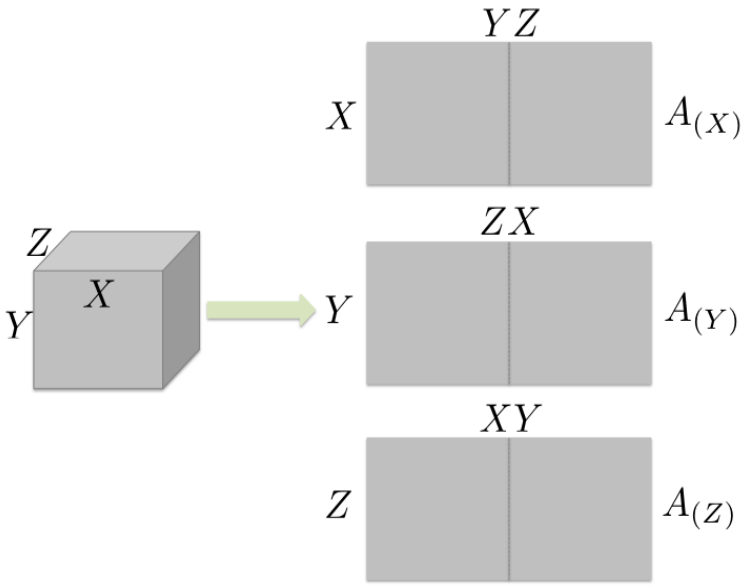


Figure 6.1: An example of matrix unfolding of a 3rd order tensor

6.2 Tensor Representation

A tensor is a high order generalization of multi-dimensional data where a vector and a matrix are regarded as a first order tensor and a second order tensor, respectively. A video contains both spatial and temporal information and can be naturally represented as a third order tensor $\in \mathbb{R}^{X \times Y \times Z}$ where X, Y, and Z are the image width, image height, and video length, respectively. Tensors can be regarded as a multilinear mapping over a set of vector spaces. Generally, useful information can be extracted using tensor decompositions, in particular, a Higher Order Singular Value Decomposition (HOSVD) [27] which has been proven to be useful in statistical data analysis [74] and dimensionality reduction [25]. The tensor algebra used in this chapter is reviewed in chapter 3.

Just as a matrix can be factorized using Singular Value Decomposition (SVD), a tensor can also be factorized using HOSVD. Let $\mathcal{A}$ be an order N tensor $\in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$. $\mathcal{A}$ can be converted to a set of matrices via a matrix unfolding operation. Matrix unfolding maps a tensor $\mathcal{A}$ to a set of matrices $A_{(1)}$, $A_{(2)}$, ..., $A_{(N)}$, where $A_{(k)} \in \mathbb{R}^{I_k \times (I_1 \times \cdots \times I_{k-1} \times I_{k+1} \cdots \times I_N)}$ is a mode-k matrix of $\mathcal{A}$. An example of matrix unfolding of a 3rd order tensor is given in Figure 6.1,

where the rows are represented by a single order of the tensor and the columns are composed by two orders of the tensor. A tensor can be unfolded N times where N corresponds to the order of the tensor.

HOSVD is a method for tensor decomposition and is the heart of many applications [53]. HOSVD first unfolds a tensor to a set of matrices and performs factorization on all unfolded matrices. Similar to matrix decomposition, the mode-k matrix $A_{(k)} \in \mathbb{R}^{n \times m}$ is factored using SVD as follows:

$$A_{(k)} = U^{(k)} \Sigma^{(k)} V^{(k)^T} \tag{6.1}$$

where $\Sigma^{(k)} \in \mathbb{R}^{n \times m}$ is a diagonal matrix, $U^{(k)} \in \mathbb{R}^{n \times n}$ is an orthogonal matrix spanning the column space of $A_{(k)}$ associated with nonzero singular values, and $V^{(k)} \in \mathbb{R}^{m \times m}$ is an orthogonal matrix spanning the row space of $A_{(k)}$ associated with nonzero singular values. Then, an N order tensor can be factorized using HOSVD as follows:

$$\mathcal{A} = \mathcal{S} \times_1 U^{(1)} \times_2 U^{(2)} \ldots \times_n U^{(N)} \tag{6.2}$$

where $\mathcal{S} \in \mathbb{R}^{(I_1 \times I_2 \times \cdots \times I_N)}$ is a core tensor, $U^{(1)}$, $U^{(2)}$, $\ldots$, $U^{(N)}$ are orthogonal matrices spanning the column space associated with nonzero singular values described in Equation (6.1), and $\times_k$ denotes the mode-k multiplication. The order of a tensor is preserved by HOSVD. Equation (6.1) and Equation (6.2) show that the HOSVD is a generalization of the matrix SVD and the variation in the mode-k matrix is captured independent to the other modes.

6.3 Product Manifolds

A product manifold can be recognized as a complex compound object in a high dimensional space composed by a set of lower dimensional objects. For example, a line whose elements y in $\mathbb{R}^1 \times$ a circle whose elements x in $\mathbb{R}^2$ becomes an infinite cylinder whose elements (x, y) in $\mathbb{R}^3$ shown in Figure 6.2. Formally, this product can be expressed as:

$$D^2 = \{x \in \mathbb{R}^2 : |x| < 1\} \tag{6.3}$$

$$I = \{y \in \mathbb{R} : |y| < 1\} \tag{6.4}$$

$$D^2 \times I = \{(x, y) \in \mathbb{R}^2 \times \mathbb{R} : |x| < 1 \quad \text{and} \quad |y| < 1\} \tag{6.5}$$

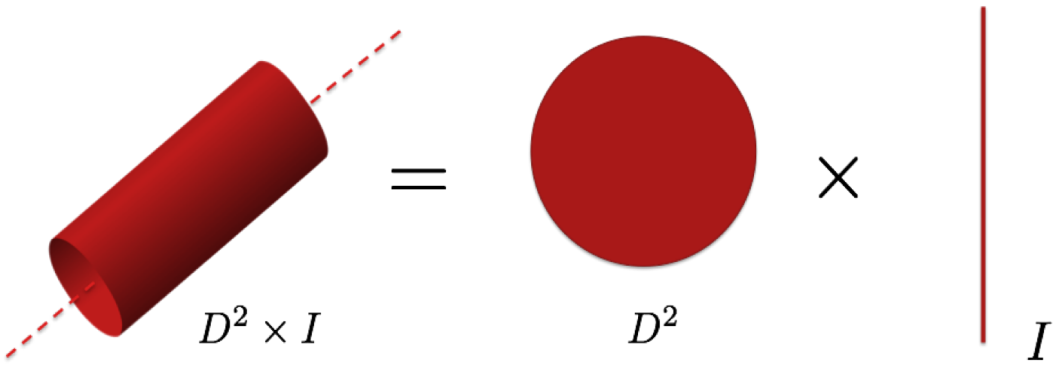


Figure 6.2: An example of an infinite cylinder: a circle cross an interval

where D^2 and I are viewed as topological spaces. Figure 6.2 reveals that a cylinder is a product of a circle and an interval where the dash line represents the concept of infinity. This open cylinder is both a circle of intervals and an interval of circles. The product manifold may be viewed as the cross section of lower dimensional objects. Formally, let $\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_q$ be a set of manifolds. The set $\mathcal{M}_1 \times \mathcal{M}_2 \times \ldots \times \mathcal{M}_q$ is called the product of the manifolds where the manifold topology is equivalent to the product topology. Thus, a product manifold is defined as:

$$\mathcal{M} = \mathcal{M}_1 \times \mathcal{M}_2 \times \cdots \times \mathcal{M}_q \tag{6.6}$$

$$= \{(x_1, x_2, \ldots, x_q) : x_1 \in \mathcal{M}_1, x_2 \in \mathcal{M}_2, \ldots, x_q \in \mathcal{M}_q\} \tag{6.7}$$

where $\times$ denotes the Cartesian product, $\mathcal{M}_k$ represents a factor manifold (a topological space), and x_k is an element in $\mathcal{M}_k$. Note that the dimension of a product manifold is the summation of all factor manifolds.

Product manifolds allow us to generalize classification problems to higher dimensional spaces. The hypothesis is that these higher dimensional spaces would yield better discriminability for classification. In our geometric framework, the embedding on product manifolds is via high order factorization.

6.3.1 Factorization in Product Spaces

As discussed in Section 6.2, HOSVD is built up from the unfolded matrices (modes) via matrix unfolding. The variation of each mode is captured by HOSVD. However, as we are about to make clear, the traditional definition of HOSVD will cause difficulties for us as we attempt to use HOSVD to relate a tensor on a product manifold.

Note that the column of every unfolded matrix $A_{(k)}$ is composed by multiple orders from the original tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$. This fact can also be observed in Figure 6.1. Let m be the dimension of the columns, $I_1 \times I_2 \times \ldots I_{k-1} \times I_{k+1} \cdots \times I_N$, and n be the dimension of the rows, I_k, for an unfolded matrix $A_{(k)}$. We can then assume that the dimension of the columns is greater than the dimension of the rows, i.e. $m > n$. This implies that the unfolded matrix $A_{(k)}$ only spans n dimensions.

According to the SVD Equation (6.1), $U^{(k)} \in \mathbb{R}^{n \times n}$ is the orthogonal matrix spanning the column space associated with nonzero singular values. Because the number of columns is larger than the number of rows, i.e., $m > n$, $U^{(k)}$ is actually a point on a special orthogonal group $\mathbb{SO}(n)$ and there is no closed-form solution for computing the geodesic distance on $\mathbb{SO}(n)$. Furthermore, the geodesic distance would always be zero when we view points on $\mathbb{SO}(n)$ as Graßmannian. This is due to the fact that Graßmannian can be represented as the quotient space of $\mathbb{SO}(n)$. In other words, we can always find a rotation matrix (a mapping) to rotate a point to the other in $\mathbb{SO}(n)$. As such, traditional HOSVD employed $U^{(k)}$ for factorization is not the appropriate choice to form a product manifold.

On the other hand, $V^{(k)}$ in Equation (6.1) is the orthogonal matrix spanning the row space and it only spans n dimensions because $m > n$. Therefore, $V^{(k)}$ can be represented by an $m \times n$ orthogonal matrix and it is a point on a Graßmann manifold. This observation motivates us to form a product manifold by modifying the HOSVD.

The modification for the existing HOSVD is simple. Since the $V^{(k)}$ is the orthogonal matrix spanning the row space, all we need is to make it span the column space because a point on a Graßmann manifold represents a subspace spanned by the column space of an orthogonal matrix. To do so, we can simply take the $V^{(k)}$ from Equation (6.1). Alternatively, we can change

the matrix unfolding of $A_{(k)}$ from $\mathbb{R}^{I_k \times (I_1 \times \cdots \times I_{k-1} \times I_{k+1} \cdots \times I_N)}$ to $\mathbb{R}^{(I_1 \times \cdots \times I_{k-1} \times I_{k+1} \cdots \times I_N) \times I_k}$. Therefore, we need to transpose the unfolded matrix and the modified HOSVD can then be written as:

$$\mathcal{A} = \hat{\mathcal{S}} \times_1 V^{(1)} \times_2 V^{(2)} \ldots \times_N V^{(N)} \tag{6.8}$$

where the dimension of the core tensor $\hat{S}$ is $\mathbb{R}^{(I_2 \times I_3 \times \cdots \times I_N) \times (I_1 \times I_3 \cdots \times I_N) \times \cdots \times (I_1 \times I_2 \times \ldots I_{(N-1)})}$. One can easily verify that the core tensor $\hat{\mathcal{S}}$ along with $V^{(k)}$ would perfectly reconstruct the tensor $\mathcal{A}$.

Because $V^{(k)}$ is an orthogonal matrix, every such matrix has an associated point on a Graßmann manifold. Furthermore, a set of orthogonal matrices $V^{(1)}$, $V^{(2)}$, $\ldots$, $V^{(N)}$ forms a set of Graßmann manifolds $\mathcal{M}_1$, $\mathcal{M}_2$, $\ldots$, $\mathcal{M}_N$ where the dimension of each factor manifold is different. In other words, $V^{(k)}$ is the component for a product manifold.

Interestingly, using the modified HOSVD, we are able to factorize each order of the tensor into a factor manifold. Each factor manifold spans one order of a tensor in a column space (Recall that every point on an $\mathcal{G}_{n,p}$ represents a subspace spanned by the column space) whereas the traditional HOSVD spans multiple factors. Hence, the modified HOSVD can have a one-to-one factorization between the order of a tensor and a factor manifold.

6.4 Graßmann Product Manifolds

To exploit the idiosyncratic nature of the geometry of visual data, we choose Graßmann manifolds as the factor manifolds because the space of Graßmannian is well-defined, in particular, there are closed-form geodesic distances.

A product manifold composed by a set of Graßmann manifolds is called Graßmann Product Manifolds (GPM) described as:

$$\mathcal{M} = \mathcal{G}_1 \times \mathcal{G}_2 \times \cdots \times \mathcal{G}_N \tag{6.9}$$

where $\mathcal{G}_k$ is a factor manifold which is a Graßmann manifold in our case. The Graßmann manifold $\mathcal{G}_k$ consists of a sets of I_k-dimensional linear subspaces embedded in

$\mathbb{R}^{(I_1 \times I_2 \ldots I_{k-1} \times I_{k+1} \ldots I_{N-1} \times I_N)}$. As such, the Graßmann product manifold represents the Cartesian product space of Graßmannian.

The topology of a product manifold is expressed as the Cartesian product of lower dimensional manifolds. The composition law allows a product manifold to be topologically decoupled and the intrinsic elements can be computed on each factor manifolds. This decoupling provides a means for computing the geodesic distance on a product manifold.

6.4.1 Geodesic Distance on Graßmann Product Manifolds

The space of a Graßmann manifold is curved, and the shortest path between two points on a manifold is geodesic. Moreover, the space of a product manifold composed by a set of Graßmann manifolds is also curved. Thus, the distance between two points on a product manifold should also be geodesic.

It is known that the geodesic in a product manifold $\mathcal{M}$ is the product of geodesics in $\mathcal{M}_1$, $\mathcal{M}_2$, $\ldots$, $\mathcal{M}_N$ [68, 8]. As such, for any differentiable curve γ parametrized by t, we have $\gamma(t) = (\gamma_i(t), \gamma_j(t))$ where γ_i and γ_j are the geodesics on $\mathcal{M}_i$ and $\mathcal{M}_j$ respectively. From this observation, a geodesic distance on a product manifold can be formulated as:

$$d_{\mathcal{M}}(\mathcal{A}, \mathcal{B}) = \| \sin \Theta \|_2 \tag{6.10}$$

where $\mathcal{A}$ and $\mathcal{B}$ are N order tensors, and $\Theta = (\theta_1, \theta_2, \ldots, \theta_N)$ where $\theta_k \in \mathcal{G}_k$ is a list of canonical angles and are computed separately on each factor (Graßmann) manifold. The geodesic distance on GPM is defined as chordal distance [23]. Nevertheless, other distance metrics [28] can also be considered. As Equation (6.10) shows, the canonical angles on product manifolds can be expressed as a Cartesian product of canonical angles computed by factor manifolds. Consequently, the geodesic distance is formulated on product manifolds.

This development of geodesic distance on the product manifold can be related back to our cylinder example where a circle in $\mathbb{R}^2$ and a line in $\mathbb{R}^1$ form an open cylinder in $\mathbb{R}^3$ in a product space. Recall that a Grassmann manifold is a set of p-dimensional linear subspaces. In analogous fashion, the product of a set of $p_1, p_2, \ldots, p_N$ linear subspaces forms a set of product subspaces whose dimension is $(p_1 + p_2 + \ldots + p_N)$. The product subspaces are the elements on a product

manifold. This observation is consistent with the Θ in Equation (6.10) where the number of canonical angles agrees with the dimension of product subspaces on the product manifold.

Note that canonical angles θ_k are measured between $V_{\mathcal{A}}^{(k)}$ and $V_{\mathcal{B}}^{(k)}$ where each is an orthogonal matrix spanning the row space associated with nonzero singular values from a mode-k matrix. As such, an N order tensor in $\mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ would span N row spaces in I_1, I_2, ..., I_N, respectively, and the dimension of product subspaces on a product manifold is the sum of each order of a tensor, i.e., ($\sum_{i=1}^{N} = I_1 + I_2 + \ldots + I_N$).

6.5 Classification on Graßmann Product Manifolds

Putting all the components together for video classification, we employ a simple nearest neighbor classifier. Let $\mathcal{A}$ and $\mathcal{B}_j$ be 3rd order tensors where $\mathcal{A}$ is a query video and $\mathcal{B}_j$ is a target video. Then, the classification can be performed as follows:

$$j^* = \underset{j \in \text{target}}{\operatorname{argmin}} \, d_{\mathcal{M}}(\mathcal{A}, \mathcal{B}_j) \tag{6.11}$$

The tensor representation on a product manifold for a video models the variations in both space and time. Each video is explicitly formulated as multiple effects and the geometry of the space is properly considered. The geodesic distance on a product manifold is not only geometrically sound, but is also very useful.

6.6 Experimental Results

We will test our method on the Cambridge-Gesture database [50][1] and the KTH human action database [91][2]. The Cambridge-Gesture database includes nine types of gestures taken under five different illuminations. The video frame size is 320×240 and video lengths are diverse. The KTH human action database has six categories of human actions and is the largest action

[1] ftp://mi.eng.cam.ac.uk/pub/CamGesData/

[2] http://www.nada.kth.se/cvap/actions/

	GPM	GM	TCCA [48]	DCCA [49]
Set1	89%	77%	81%	63%
Set2	86%	70%	81%	61%
Set3	89%	76%	78%	65%
Set4	87%	77%	86%	69%
Average	88%	75%	82%	65%

Table 6.1: Classification rates for gesture action classification on the Cambridge-Gesture database

data set publicly available. Each video frame is scaled to 160×120 and the number of frames for each video sequence also varies.

For comparison purpose, we also view a video as a set of images. This set of images is projected on a Graßmann manifold and the geodesic distance between image-sets is computed for action classification. We call this method Graßmann Manifolds (GM). This baseline algorithm contrasts the merit of the geometry between a tensor on a product manifold and a tensor on a Graßmann manifold.

6.6.1 Gesture Action Classification

The Cambridge-Gesture database includes 900 video sequences, 100 for each of nine gestures. Each of the 100 videos per gesture class is further broken down into five illuminations (Set1, Set2, Set3, Set4, and Set5) and ten motions from each of two subjects. Examples of these gestures are given in Figure 6.3.

All video sequences are resized to $20 \times 20 \times 32$. Note that our method can handle varied video lengths, but the standardized procedure is simply for computational convenience. To standardize the video length, we collect the middle 32 frames from a video sequence. Furthermore, no space-time alignment is performed on this data set.

Following the experimental protocol of [48], the data set is partitioned into a number of illumination sets where Set1, Set2, Set3, and Set4 are the test sets, and Set5 is the training set. Furthermore, the training set is randomly divided into training and validation sets (10 sequences for training and the other 10 sequences for validation). Since we do not perform prior training, we discard the validation set.

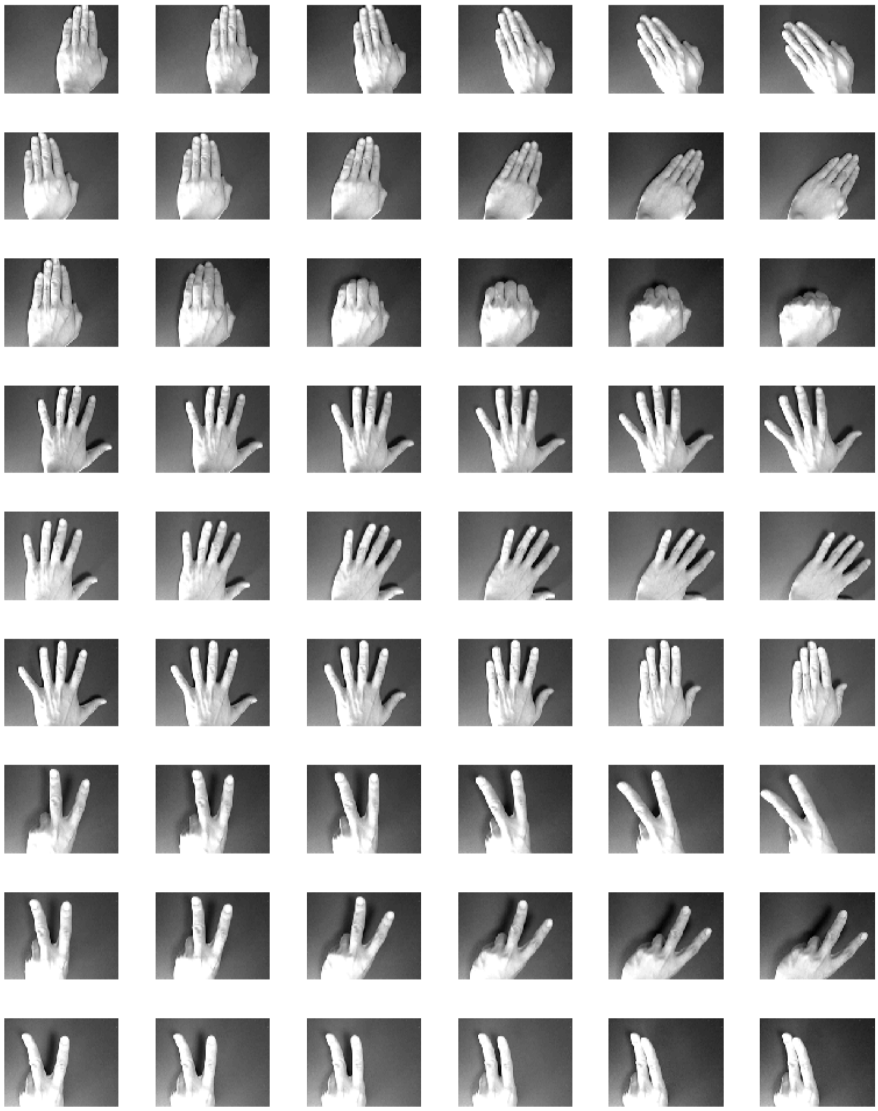


Figure 6.3: Hand gesture action samples. Each row depicts a class of hand gesture actions. (Flat-Leftward (FL), Flat-Rightward (FR), Flat-Contract (FC), Spread-Leftward (SL), Spread-Rightward (SR), Spread-Contract (SC), V-Shape-Leftward (VL), V-Shape-Rightward (VR), and V-Shape-Contract (VC))

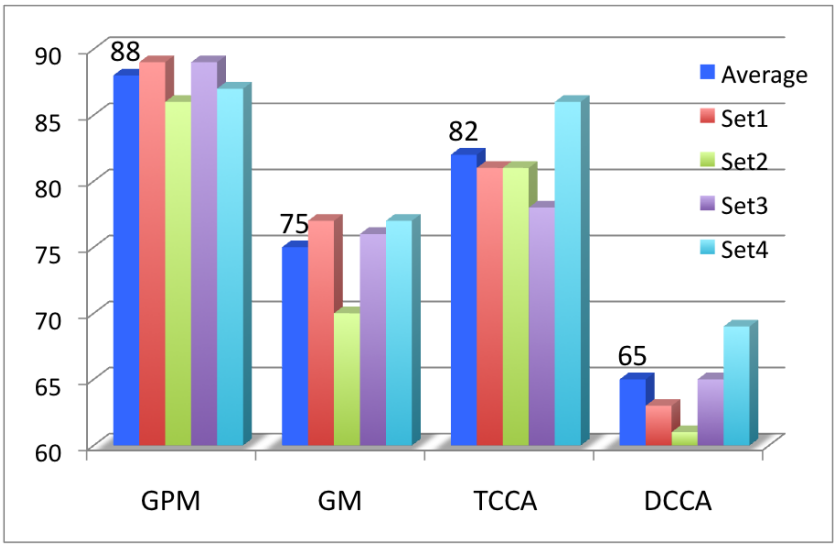


Figure 6.4: The bar chart for gesture action classification on the Cambridge-Gesture database

The gesture action classification results are reported in Table 6.1 and our method (GPM) outperforms the baseline algorithm, Graßmann manifold (GM), and the current state-of-the-art methods, tensor CCA (TCCA) [48] and discriminative CCA (DCCA) [49], on all illumination data sets[3]. As Table 6.1 suggests, the GPM is able to capture more discriminative information from videos than GM where the GM uses the spatial but ignores the temporal information. The gesture action classification results are also depicted as a bar chart given in Figure 6.4 where the average accuracy is highlighted in the blue color. The key distinction between these methods is that the GPM and TCCA account for all orders of a tensor while the GM and DCCA only consider the tensor as an image set. In addition, both TCCA and DCCA require prior training data whereas the GPM and GM are non-trained methods.

The classification results for GPM and TCCA are further divided into categories and presented in confusion matrices in Figure 6.5. Each cell in the confusion matrix is the average classification rate from four illumination sets. The confusion matrices show that our method and

[3]We do not include Wong and Cipolla's results [116] here because their results were reported using the leave-one-out cross validation protocol.

GPM

	FL	FR	FC	SL	SR	SC	VL	VR	VC
FL	78	0	0	1	0	15	4	0	3
FR	0	84	0	0	1	1	0	13	1
FC	0	0	84	0	0	4	0	0	13
SL	1	0	0	90	0	1	8	0	0
SR	0	4	0	0	89	0	0	8	0
SC	0	0	0	0	0	83	0	0	18
VL	0	0	0	0	0	0	96	0	4
VR	0	1	0	0	0	0	0	95	4
VC	0	0	0	0	0	9	0	0	91

TCCA [48]

	FL	FR	FC	SL	SR	SC	VL	VR	VC
FL	94	0	0	4	0	0	1	0	0
FR	0	98	0	2	0	0	0	0	0
FC	1	0	81	0	0	13	0	0	5
SL	3	0	0	95	0	0	2	0	0
SR	0	14	0	0	84	0	0	2	0
SC	5	0	0	2	0	93	0	0	0
VL	6	0	0	14	0	0	81	0	0
VR	1	17	0	1	10	0	4	68	0
VC	2	0	13	0	0	14	2	1	68

Figure 6.5: Confusion matrices for gesture action classification

TCCA handle individual gestures differently, with GPM doing better on five actions (FC, SR, VL, VR, and VC) and TCCA doing better on four actions (FL, FR, SL, and SC). Furthermore, the worst TCCA performance for a gesture is 68%, compared to 78% for our method. The best performance of TCCA for a gesture is 98%, compared to 96% for our method. Hence, GPM has better overall recognition results.

6.6.2 Human Action Classification

The KTH human action data set [91] has six types of human actions including walking, running, jogging, boxing, handwaving, and handclapping. Examples are shown in Figure 6.6. Each type of human actions is performed by 25 people with four different scenarios: outdoors (s1), outdoors with scale variation (s2), outdoors with different clothes (s3), and indoors (s4). Similar to Kim and Cipolla's settings [48], we first perform space-time alignment (location and frame cropping) on the human action videos manually[4]. Next, all video sequences are resized to $20 \times 20 \times 32$. In order to standardize to a length of 32 frames, we take the middle 32 for longer

[4]The action is repeated several times on each original video.

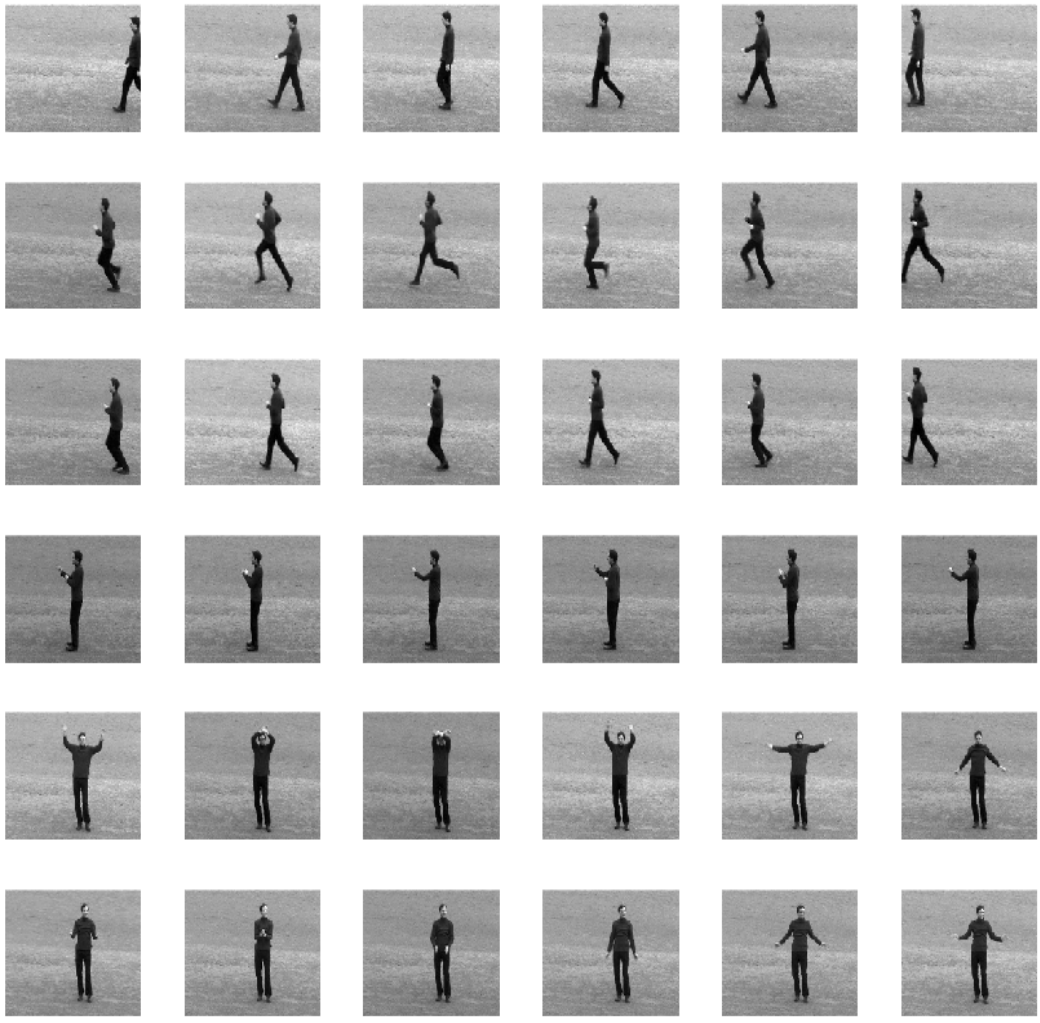


Figure 6.6: Human action samples. Rows from top to bottom are examples of walking, running, jogging, boxing, handwaving and handclapping

Average Accuracy = 96.0%

	Walk	Jog	Run	Box	Hclp	Hwav
Walk	97.9	0	2.1	0	0	0
Jog	0	100	0	0	0	0
Run	4.8	2.1	91.0	0	2.1	0
Box	0	0	0	97.2	2.8	0
Hclp	0	0	0	0	100	0
Hwav	0	0	0	0.7	10.4	89.6

Average Accuracy = 84.5%

	Walk	Jog	Run	Box	Hclp	Hwav
Walk	63.2	34.7	2.1	0	0	0
Jog	0.7	94.4	4.9	0	0	0
Run	0	39.6	58.3	2.1	0	0
Box	0	0	0	100	0	0
Hclp	0	0	0	1.4	97.2	1.4
Hwav	0	0	0	1.4	4.9	93.8

Average Accuracy = 91.8%

	Walk	Jog	Run	Box	Hclp	Hwav
Walk	99.0	1.0	0	0	0	0
Jog	4.0	89.0	7.0	0	0	0
Run	1.0	19	80.0	0	2.8	0
Box	0	0	0	97.0	0	3.0
Hclp	0	0	0	5.0	95.0	0.7
Hwav	0	0	0	0	9.0	91.0

Average Accuracy = 71.7%

	Walk	Jog	Run	Box	Hclp	Hwav
Walk	83.8	16.2	0	0	0	0
Jog	22.9	60.4	16.7	0	0	0
Run	6.3	38.9	54.9	0	0	0
Box	0.7	0	0	97.9	0.7	0.7
Hclp	1.4	0	0	35.4	59.7	3.5
Hwav	0.7	0	0	20.8	4.9	73.6

Figure 6.7: Confusion matrices for human action classification (Schüldt's protocol): Top Left (GPM), Top Right (GM), Bottom Left (BOF + SVM) [57], Bottom Right (LF + SVM) [91]

sequences, and recycle frames for videos shorter than 32 frames.

To facilitate comparison with prior work, performance on human action classification is evaluated using two protocols. The first is proposed by Schüldt et al. [91]. The KTH human action data set is divided into three subsets with different people: training set (8 persons), validation set (8 persons), and test set (9 persons). Like the gesture experiment, we discard the validation set because no prior training is required for the proposed method. The training set is further divided into four groups of {s1}, {s1, s4}, {s1, s3, s4}, and {s1, s2, s3, s4}. The test set is always {s1, s2, s3, s4}.

The results are presented in the confusion matrices shown in Figure 6.7. Each cell in the confusion matrix is the average result from the four training groups. The results of GPM, GM, the bag-of-features SVM (BOF + SVM) [57] and the local feature SVM (LF+SVM) [91] are presented in the top left confusion matrix, top right confusion matrix, bottom left confusion matrix, and bottom right confusion matrix, respectively. As Figure 6.7 demonstrates, the GPM improves

	GPM	GM	TCCA [48]	DCCA [49]	STIP [116]	STW [78]
Walking	98%	75%	99%	100%	88%	82%
Jogging	99%	97%	90%	80%	75%	53%
Running	97%	70%	88%	68%	77%	88%
Boxing	97%	99%	98%	97%	92%	98%
Handclapping	98%	95%	100%	99%	100%	86%
Handwaving	95%	100%	97%	99%	88%	93%
Average	97%	89%	95%	90%	87%	83%

Table 6.2: Classification results for human action classification on the KTH human action database (Leave-one-out cross validation)

the classification accuracies in jogging, running, and handclapping. Overall, the GPM achieves 96% average classification rate whereas the GM achieves 84.5%, the BOF+SVM obtains 91.8% and the LF+SVM gets 71.7%.

The second experiment protocol, used by [48, 116, 78], is the leave-one-out (LOO) cross validation. The classification results using LOO for the KTH human actions are reported in Table 6.2. The first thing to note in Table 6.2 is that no algorithm is universally best. In terms of top classification rates, STIP, DCCA, TCCA, GM, and GPM has the best recognition result among the six actions. However, when the GPM is better, it is typically by a larger amount. This is due to the utilization of underlying geometry on a product manifold and is reflected in the higher overall average classification rate of 97% versus 95% for TCCA and 90% for DCCA.

Looking at the results for both protocols on the KTH human action data set, GPM achieves the highest overall classification rate. Interestingly, it is notable that the two actions most commonly confused by the other approaches [57, 91, 48, 49, 116, 78] are jogging and running, and GPM is able to reduce this ambiguity greatly.

Another interesting observation is between GPM and GM. GPM is a product manifold composed by a set of Graßmann manifolds whereas GM is a single Graßmann manifold. Both Figure 6.7 and Table 6.2 reveal that GPM significantly outperforms GM in the walking and running categories. This improvement is due in part to the tensor representation and the use of the underlying geometry of product manifolds where both spatial and temporal information are exercised.

6.7 Discussion

The methods tested in this chapter can be logically divided into two categories. They are feature-based methods [57, 91, 116, 78], and pixel-based methods [48, 49]. The proposed approach is pixel-based. Conceptually, pixel-based methods are simpler because they do not need additional human intervention and/or machine learning algorithms to identify the features, and a feature detector to locate the landmarks. Therefore, they are arguably easier to apply to new action classification problems.

All the algorithms that we compared against in Section 6.6 require prior training. Note the LOO protocol in particular, as exemplified by the results in Table 6.2, works strongly in favor of highly trained methods by maximizing the available training data. Of course, in practice large amounts of training data are not always available. Whenever training is utilized, the opportunity arises for performance to degrade if there is a mismatch between training and operational data. Because our method depends upon the intrinsic geometry of the videos expressed through product manifolds, no prior training is involved.

Not only is our method generic insofar as no parameters need tuning, but also the matching time is fast. With a non-optimized MATLAB implementation, each match between two videos takes about eight milliseconds on a standard modern computer. Furthermore, our geometric framework is not limited to 3rd order tensors but generalizable to N-way tensors. Thus, the potential applications of our geometric framework would go beyond video classification.

On the other hand, the proposed method inherits a drawback from pixel-based approaches. Like every pixel-based method, the proposed approach is sensitive to cluttered backgrounds. In cases where cluttered backgrounds arise, a preprocessing step like video segmentation may be needed to extract the motion of interest. Nevertheless, the focus of this search is the geometric framework relating a tensor on a product manifold and video segmentation is outside the scope of this dissertation.

Chapter 7

Conclusions

Due to the nonlinearity of image spaces, Euclidean geometry is not sufficient to characterize the variability among images. In this dissertation, novel geometric frameworks for visual recognition problems are presented. Our geometric frameworks are developed based upon special manifolds in which the underlying geometry is a curved surface embedded in a high dimensional Euclidean space. The special manifolds are mathematically well-defined and geometrically well-understood. By properly choosing a parameter space, the idiosyncratic aspects of visual data are exploited for pattern recognition. Theoretical considerations from differential geometry have led in this dissertation to practical frameworks for performing state-of-the-art face recognition and action classification.

The theme of this dissertation is the development of geometric frameworks for visual recognition and the demonstration of the significance of the geometry of space. Three major research results have been described including Canonical Stiefel Quotient (CSQ), Graßmann Registration Manifolds (GRM), and Graßmann Product Manifolds (GPM). GRM and CSQ bridge the gap from a single still image to image-set by perturbing an image in registration and illumination spaces. GPM models a video as a 3rd order tensor and relates the tensor representation to a product manifold.

CSQ exploits the underlying geometry of Stiefel manifolds for generic face recognition in illumination spaces. An illumination model is used to relight a single image to a set of illumination variants. These illumination variants are adopted to form two projections on a tangent space of a Stiefel manifold. The magnitudes of these projections are measured by an appropriate

canonical metric. The CSQ is computed as a ratio between the canonical metrics of projections.

Our experimental results on CMU-PIE and YaleB datasets conclude that CSQ not only performs well but is robust to the choice of training sets as well. The direct comparison between the CGHP (the image-set version of a novelty filter) and the standard novelty filter assures the benefits of image set matching. Furthermore, CSQ is not restricted to a particular illumination model. Instead, any illumination model, or even any general systematic means of obtaining image sets, could fruitfully employ the CSQ. distance measure.

GRM perturbs a registration manifold using affine transformations and forms a tangent space from a local neighborhood. Since the tangent space admits a vector space structure, it is embedded on a Graßmann manifold. The geodesic distance is exploited for face recognition. Unlike many manifold learning algorithms, GRM does not require dense samples or training data. Therefore, GRM is more generic and flexible. Empirical evidence suggests that approximately 100 local registration variants from the affine registration manifold optimizes recognition performance. To reduce the computational burden, a coarse to fine matching strategy is introduced that allows GRM to match a small set of gallery candidates.

Using whole face images, i.e. a holistic representation, our proposed GRM algorithm does very well on the FERET tests. Specifically, the rank one identification rates on the $Dup1$ and $Dup2$ probe sets are the highest for any holistic method. Performance is further boosted with the introduction of local features. In addition, among all the non-trained algorithms, our algorithm achieves the best result of the FERET $Dup2$ data set, which is generally considered the most difficult data set on FERET.

GPM further demonstrates the value of taking into account the underlying geometry in video classification. We represent a video as a 3rd order tensor and map it to a product manifold where each factor manifold is a Graßmannian. The realization of points on these Graßmannians is achieved by applying a modified HOSVD to a tensor representation of the action video. A natural metric is inherited from the factor manifolds since the geodesic on the product manifold is given by the product of the geodesic on the Graßmann manifolds.

This composite geodesic distance is formulated and applied to the problem of action clas-

sification. Experimental results show that GPM performs very well on the Cambridge gesture and KTH human-action data sets. Finally, GPM is generic in the sense that no prior training is required and the matching time is fast.

The geometric frameworks based on special manifolds developed in this dissertation promote the advantages of the geometry of special manifolds and provide a new perspective for pattern classification. The fundamental idea of these frameworks is to view a set of images as a point in some parameter space and perform classification according to the geometry of the chosen space. Remarkable results on both still images and videos have been achieved for the applications of face recognition, gesture recognition, and human action classification. Our introduction of geometric frameworks on special manifolds further advances the research on analytic manifolds for visual recognition.

7.1 Future Work

The results reported in this dissertation encourage further study of the role that special manifolds can play in formulating practical solutions to real-world visual recognition tasks. One possible extension will involve the integration of special manifolds with statistical inference. Machine learning may be viewed as an optimization problem in addition to some geometric constrained structure. As such, data are viewed geometrically. This may provide new insights in learning the structure of patterns.

From a geometric point of view, we can also utilize the concepts of GRM and GPM. While GRM forms a tangent space from a registration manifold, we can construct a tangent bundle, a set of tangent spaces. Consequently, the tangent bundle can be related as a point on a product manifold. We can then employ tangent bundles for classification.

Our geometric frameworks consider a set of images as a point in some parameter space. While the distance metrics are well-defined, the characterization of image sets has significantly impact on classification performance. A challenging question arises in the selection of image sets. What sets of images would yield an optimum class separation and how can we obtain those images? Further investigation in this area would definitely enrich the field of pattern recognition.

Appendix A

Derivations and Properties of Canonical Angles and Canonical Vectors

A.1 Derivations

Given a constrained optimization problem as follows:

$$\max_{W_x, W_y} \operatorname{tr}\left\{W_x^T C_{xy} W_y\right\} \tag{A.1}$$

subject to

$$\begin{aligned} W_x^T C_{xx} W_x = I \qquad &\Rightarrow \qquad ||XW_x||_F^2 = 1 \\ W_y^T C_{yy} W_y = I \qquad &\Rightarrow \qquad ||YW_y||_F^2 = 1 \end{aligned} \tag{A.2}$$

where $C_{xy} = X^T Y$, $C_{xx} = X^T X$, and $C_{yy} = Y^T Y$. We can express it as the Lagrangian formulation given by

$$L(W_x, W_y, \lambda_x, \lambda_y) = W_x^T C_{xy} W_y + \lambda_x (1 - W_x^T C_{xx} W_x) + \lambda_y (1 - W_y^T C_{yy} W_y) \tag{A.3}$$

where λ_x and λ_y are the Lagrangian multipliers.

For convenient, we divide the λ by 2, and we have

$$L(W_x, W_y, \lambda_x, \lambda_y) = W_x^T C_{xy} W_y + \frac{\lambda_x}{2} (1 - W_x^T C_{xx} W_x) + \frac{\lambda_y}{2} (1 - W_y^T C_{yy} W_y) \tag{A.4}$$

Taking derivatives with respect to W_x and W_y, we obtain

$$\frac{\partial L}{\partial W_x} = C_{xy}W_y - \lambda_x C_{xx}W_x = 0 \tag{A.5}$$

$$\frac{\partial L}{\partial W_y} = C_{yx}W_x - \lambda_y C_{yy}W_y = 0 \tag{A.6}$$

Note that C_{yx} = C_{xy}^T

Premultiplying W_x^T to Equation (A.5), we have

$$W_x^T C_{xy}W_y - \lambda_x W_x^T C_{xx}W_x = 0 \tag{A.7}$$

Premultiplying W_y^T to Equation (A.6), we get

$$W_y^T C_{yx}W_x - \lambda_y W_y^T C_{yy}W_y = 0 \tag{A.8}$$

Since the trace operator induces transpose invariant, we have

$$(W_x^T C_{xy}W_y)^T = W_y^T C_{yx}W_x \tag{A.9}$$

Subtracting Equation (A.8) from Equation (A.7), we obtain

$$\lambda_y W_y^T C_{yy}W_y - \lambda_x W_x^T C_{xx}W_x = 0 \tag{A.10}$$

$$\lambda_x W_x^T C_{xx}W_x = \lambda_y W_y^T C_{yy}W_y \tag{A.11}$$

From Equation (A.2), we know $W_x^T C_{xx}W_x$ = 1, and $W_y^T C_{yy}W_y$ = 1, then we obtain

$$\lambda_x = \lambda_y = \lambda \tag{A.12}$$

Substituting Equation (A.12) into $\frac{\partial L}{\partial W_x}$ and $\frac{\partial L}{\partial W_y}$, we have

$$C_{xy}W_y - \lambda C_{xx}W_x = 0 \tag{A.13}$$

$$C_{yx}W_x - \lambda C_{yy}W_y = 0 \tag{A.14}$$

From Equation (A.13), we obtain

$$C_{xy}W_y = \lambda C_{xx}W_x \tag{A.15}$$

$$W_y = \lambda C_{xy}^{-1} C_{xx}W_x \tag{A.16}$$

Substituting W_y into Equation (A.14), we have

$$C_{yx}W_x = \lambda C_{yy}W_y \tag{A.17}$$

$$= \lambda C_{yy}(\lambda C_{xy}^{-1}C_{xx}W_x) \tag{A.18}$$

$$= \lambda^2 C_{yy}C_{xy}^{-1}C_{xx}W_x \tag{A.19}$$

Assuming non-singularity of C_{yy},

$$(C_{yy}C_{xy}^{-1})^{-1}C_{yx}W_x = \lambda^2 C_{xx}W_x \tag{A.20}$$

$$C_{xy}C_{yy}^{-1}C_{yx}W_x = \lambda^2 C_{xx}W_x \tag{A.21}$$

Similarly,

$$C_{yx}C_{xx}^{-1}C_{xy}W_y = \lambda^2 C_{yy}W_y \tag{A.22}$$

Note that it is a generalized eigenvalue problem ($Ax = \lambda Bx$). Because the eigenvalues of the two eigen-systems are identical, we can combine Equation (A.13) and Equation (A.14) into a single generalized eigen-system given as follows.

$$\begin{bmatrix} 0 & C_{xy} \\ C_{yx} & 0 \end{bmatrix} \begin{bmatrix} W_x \\ W_y \end{bmatrix} = \lambda \begin{bmatrix} C_{xx} & 0 \\ 0 & C_{yy} \end{bmatrix} \begin{bmatrix} W_x \\ W_y \end{bmatrix} \tag{A.23}$$

Due to the fact that the matrices C_{xx} and C_{yy} are symmetric positive definite, we can utilize the Cholesky decomposition to factorize these matrices [15]. Let

$$C_{xx} = R_{xx} \cdot R_{xx}^T \tag{A.24}$$

where R_{xx} is a lower triangular matrix. Let

$$u_x = R_{xx}^T \cdot W_x \qquad \Rightarrow \qquad W_x = R_{xx}^{-T} \cdot u_x \tag{A.25}$$

Then, we can rewrite Equation (A.21) as

$$C_{xy}C_{yy}^{-1}C_{yx}W_x = \lambda^2 R_{xx}R_{xx}^T W_x \tag{A.26}$$

$$R_{xx}^{-1}C_{xy}C_{yy}^{-1}C_{yx}R_{xx}^{-T}u_x = \lambda^2 u_x \tag{A.27}$$

Likewise for the Equation (A.22), we are therefore with a symmetric eigenvalue problem ($Ax = \lambda x$).

If, on the other hand, both C_{xx} and C_{yy} are nonsingular matrices, we can directly derive Equation (A.21) and Equation (A.22) as an eigenvalue problem by taking the inverse of C_{xx} and C_{yy}, and it becomes

$$C_{xx}^{-1}C_{xy}C_{yy}^{-1}C_{yx}W_x = \lambda^2 W_x \quad (A.28)$$

$$C_{yy}^{-1}C_{yx}C_{xx}^{-1}C_{xy}W_y = \lambda^2 W_y \quad (A.29)$$

where λ^2 are the eigenvalues of $C_{xx}^{-1}C_{xy}C_{yy}^{-1}C_{yx}$ which are the maximum correlations.
The canonical angles (canonical correlations) and the canonical vectors are associated by

$$\cos(\theta_k) = \lambda_k, \qquad U = XW_x, \qquad V = YW_y \quad (A.30)$$

where λ_k are the eigenvalues, and W_x and W_y are the eigenvectors of the generalized eigenvalue problem. Note that maximizing the canonical correlations is equivalent to minimizing the canonical angles. Furthermore, Akaike [4] showed that the mutual information between two time series is given by

$$I(T_1 \parallel T_2) = \sum_{i=1}^{p} \log\left(\frac{1}{1-\lambda_i^2}\right) \quad (A.31)$$

where T_1 and T_2 are time sequences.

A.2 Properties

A.2.1 Invariance to Linear Transformations

One important property is that canonical angles are invariant to linear transformations of data matrices [16]. Let X and Y be the mean centered data matrices whose dimensionality is $n \times p$ and $n \times q$, respectively. Consider two nonsingular projection matrices U and V whose dimensionality is also $n \times p$ and $n \times q$, respectively. Then, a linear transform can be written as

$$\hat{X} = U^T X \quad (A.32)$$

$$\hat{Y} = V^T Y \quad (A.33)$$

Then, the sample covariance matrices can be expressed as

$$\hat{C}_{xx} = \hat{X}\hat{X}^T = U^T X (U^T X)^T = U^T X X^T U = U^T C_{xx} U \tag{A.34}$$

$$\hat{C}_{yy} = \hat{Y}\hat{Y}^T = V^T Y (V^T Y)^T = V^T Y Y^T V = V^T C_{yy} V \tag{A.35}$$

$$\hat{C}_{xy} = \hat{X}\hat{Y}^T = U^T X (V^T Y)^T = U^T X Y^T V = U^T C_{xy} V \tag{A.36}$$

$$\hat{C}_{yx} = \hat{Y}\hat{X}^T = V^T Y (U^T X)^T = V^T Y X^T U = V^T C_{yx} U \tag{A.37}$$

According to Equation (A.30), we have two generalized eigen-system written as

$$\hat{C}_{xy}\hat{W}_y = \hat{\lambda}\hat{C}_{xx}\hat{W}_x \tag{A.38}$$

$$\hat{C}_{yx}\hat{W}_x = \hat{\lambda}\hat{C}_{yy}\hat{W}_y \tag{A.39}$$

Substituting Equation (A.34), Equation (A.35), Equation (A.36), and Equation (A.37) into Equation (A.38) and Equation (A.39), we have

$$U^T C_{xy} V \hat{W}_y = \hat{\lambda} U^T C_{xx} U \hat{W}_x \tag{A.40}$$

$$V^T C_{yx} U \hat{W}_x = \hat{\lambda} V^T C_{yy} V \hat{W}_y \tag{A.41}$$

Premultiplying Equation (A.40) and Equation (A.41) by U^{-T} and V^{-T}, respectively, we obtain

$$C_{xy} V \hat{W}_y = \hat{\lambda} C_{xx} U \hat{W}_x \tag{A.42}$$

$$C_{yx} U \hat{W}_x = \hat{\lambda} C_{yy} V \hat{W}_y \tag{A.43}$$

As a matter of fact, these two equations also represent two generalized eigen-systems and can be rewritten as

$$C_{xy} W_y = \hat{\lambda} C_{xx} W_x \tag{A.44}$$

$$C_{yx} W_x = \hat{\lambda} C_{yy} W_y \tag{A.45}$$

where

$$W_x = U\hat{W}_x \quad \Rightarrow \quad \hat{W}_x = U^{-1} W_x \tag{A.46}$$

$$W_y = V\hat{W}_y \quad \Rightarrow \quad \hat{W}_y = V^{-1} W_y \tag{A.47}$$

Then, the canonical variates are

$$\hat{W}_x^T \hat{X} = (U^{-1}W_x)^T U^T X = W_x^T U^{-T} U^T X = W_x^T X \quad \text{(A.48)}$$

$$\hat{W}_y^T \hat{Y} = (V^{-1}W_y)^T V^T Y = W_y^T V^{-T} V^T Y = W_y^T Y \quad \text{(A.49)}$$

Equation (A.48) and Equation (A.49) show that the canonical variates are identical. It means that all the necessary components would remain unchanged described as follows.

$$\hat{W}_x^T \hat{X}(\hat{W}_x^T \hat{X})^T = W_x^T X(W_x^T X)^T = W_x^T C_{xx} W_x \quad \text{(A.50)}$$

$$\hat{W}_y^T \hat{Y}(\hat{W}_y^T \hat{Y})^T = W_y^T Y(W_y^T Y)^T = W_y^T C_{yy} W_y \quad \text{(A.51)}$$

$$\hat{W}_x^T \hat{X}(\hat{W}_y^T \hat{Y})^T = W_x^T X(W_y^T Y)^T = W_x^T C_{xy} W_y \quad \text{(A.52)}$$

Thus, transformed versions of canonical correlations would be the same as the original canonical correlations such that $\hat{\lambda} = \lambda$.

A.2.2 Orthogonality of Generalized Eigenvectors

Another important property is that any pairs of canonical vectors belonging to different canonical correlation is uncorrelated. Following the derivations from [16], we consider a generalized eigenvalue problem as

$$AW = \lambda BW \quad \text{(A.53)}$$

where A and B are symmetrical matrices whose dimensionality is $n \times n$. Now, let us consider a single eigenvector, w_k, whose dimensionality is $n \times 1$ and we have

$$Aw_i = \lambda_i B w_i \quad \text{(A.54)}$$

$$Aw_j = \lambda_j B w_j \quad \text{(A.55)}$$

Premultiplying Equation (A.54) and Equation (A.55) by w_j^T and w_i^T, respectively, we obtain

$$w_j^T A w_i = \lambda_i w_j^T B w_i \quad \text{(A.56)}$$

$$w_i^T A w_j = \lambda_j w_i^T B w_j \quad \text{(A.57)}$$

Since $w_i^T A w_j$ is a scalar and A and B are symmetrical, we can transpose it such that $(w_i^T A w_j)^T$ = $w_j^T A w_i$.

$$w_i^T A w_j = \lambda_i w_i^T B w_j \tag{A.58}$$

$$w_i^T A w_j = \lambda_j w_i^T B w_j \tag{A.59}$$

Subtracting Equation (A.58) from Equation (A.59), it becomes

$$(\lambda_j - \lambda_i) w_i^T B w_j = 0 \tag{A.60}$$

Because λ_i and λ_j are distinct, $w_i^T B w_j$ must be equal to zero. Likewise, we can move the λ to the left side and we have

$$(\frac{1}{\lambda_j} - \frac{1}{\lambda_i})(w_i^T A w_j) = 0 \tag{A.61}$$

and, due to the same fact, $w_i^T A w_j$ is also equal to zero. From this observation, we get

$$w_{xi}^T C_{xx} w_{xj} = 0 \tag{A.62}$$

$$w_{yi}^T C_{yy} w_{xj} = 0 \tag{A.63}$$

$$w_{xi}^T C_{xy} w_{yj} = 0 \tag{A.64}$$

$$w_{yi}^T C_{yx} w_{xj} = 0 \tag{A.65}$$

where index i is not equal to index j. This property demonstrates the orthogonality of the canonical vectors.

A.2.3 Canonical Vectors as Linear Combinations of Data Matrices

In terms of canonical vectors, we note that $W_x \in$ span(X) and $W_y \in$ span(Y) [73]. Recall that the solution of Equation (A.1) can be found in a generalized eigenvalue problem. Let us consider the following generalized eigenvalue equations.

$$AW = \lambda BW \tag{A.66}$$

$$\begin{bmatrix} 0 & X^T Y \\ Y^T X & 0 \end{bmatrix} \begin{bmatrix} W_x \\ W_y \end{bmatrix} = \lambda \begin{bmatrix} X^T X & 0 \\ 0 & Y^T Y \end{bmatrix} \begin{bmatrix} W_x \\ W_y \end{bmatrix} \tag{A.67}$$

If we factorize a matrix B, we get

$$B = E\Lambda E^T = \sum_{i=1}^{p+q} \lambda_i e_i e_i^T \tag{A.68}$$

where E is an orthonormal matrix, e_i are the eigenvectors of B, and Λ is a diagonal matrix consisting of the corresponding eigenvalues. Let

$$E = \begin{bmatrix} E_x \\ E_y \end{bmatrix} \tag{A.69}$$

then, we have

$$BE = \begin{bmatrix} X^T X & 0 \\ 0 & Y^T Y \end{bmatrix} \begin{bmatrix} E_x \\ E_y \end{bmatrix} = \begin{bmatrix} X^T X E_x \\ Y^T Y E_y \end{bmatrix} \tag{A.70}$$

Writing it as an eigen-equation, we have

$$BE = \Lambda E \tag{A.71}$$

$$\begin{bmatrix} X^T(XE_x) \\ Y^T(YE_y) \end{bmatrix} = \Lambda \begin{bmatrix} E_x \\ E_y \end{bmatrix} \tag{A.72}$$

Equation (A.72) indicates that E_x and E_y are linear combinations of X and Y, respectively. Assuming B is a nonsingular matrix, then applying Equation (A.68), we obtain

$$B^{-1}AW = \lambda W \tag{A.73}$$

$$(E\Lambda^{-1}E^T)AW = \lambda W \tag{A.74}$$

$$E(\Lambda^{-1}E^T AW) = \lambda W \tag{A.75}$$

Equation (A.75) indicates that W is a linear combination of E. Since E is linear combinations of data matrices X and Y, this implies that W_x and W_y lie in the span of X and Y, in other words, $W_x \in$ span(X) and $W_y \in$ span(Y).

REFERENCES

[1] P.-A. Absil, R. Mahony, and R. Sepulchre. Optimization algorithms on matrix manifolds. 2008. Princeton University Press.

[2] E. Acar and B. Yener. Unsupervised multiway data analysis: A literature survey. *IEEE Transactions on Knowledge and Data Engineering*, 21(1):6–20, January 2009.

[3] T. Ahonen, A. Hadid, and M. Pietikinen. Face recognition with local binary patterns. In *European Conference on Computer Vision, Czech Republic*, pages 469–481, 2004.

[4] H. Akaike. Canonical correlation analysis of time series and the use of an information criterion. pages 27–96, 1976. in System Identification: Advances and Case Studies, Edited by R. Mehra and D. Lainiotis.

[5] O. Arandjelović and R. Cipolla. An information-theoretic approach to face recognition from face motion manifolds. *Image and Vision Computing*, 24(6):639–647, 2006.

[6] G. BakIr, A. Gretton, M. Franz, and B. Scholkopf. Multivariate regression via stiefel manifold constraints. In *Pattern Recognition, Proceedings of the 26th DAGM Symposium, Germany*, pages 262–269, 2004.

[7] R. Basri and D. Jacobs. Lambertian reflectance and linear subspaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25:218–233, 2003.

[8] E. Begelfor and M. Werman. Affine invariance revisited. In *IEEE Conference on Computer Vision and Pattern Recognition, New York*, 2006.

[9] P. Belhumeur and D. Kriegman. What is set of images of an object under all possible lighting conditions? In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 270–277, 1996.

[10] I. Bengtsson, W. Bruzda, Åsa Ericsson, J. Åke Larsson, W. Tadej, and K. Zyczkowskif. Mutually unbiased bases and hadamard matrices of order six. *Journal of Mathematical Physics*, 48, 2007.

[11] J. R. Beveridge, B. A. Draper, J.-M. Chang, M. Kirby, H. Kley, and C. Peterson. Principal angles separate subject illumination spaces in ydb and cmu-pie. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2):351–363, 2009.

[12] J. R. Beveridge, B. A. Draper, G. H. Givens, and W. Fisher. Introduction to the statistical evaluation of face recognition algorithms. In W. Zhao and R. Chellappa, editors, *Face Processing: Advanced Modeling and Methods*. Elsevier, 2005.

[13] W. Bian and D. Tao. Harmonic mean for subspace selection. In *IEEE Conference on Pattern Recognition*, 2008.

[14] A. Bissacco, A. Chiuso, Y. Ma, and S. Soatto. Recognition of human gaits. In *IEEE Conference on Computer Vision and Pattern Recognition, Hawaii*, pages 270–277, 2001.

[15] A. Björck and G. Golub. Numerical methods for computing angles between linear subspaces. *Mathematics of Computation*, pages 579–594, 1973.

[16] M. Borga. Learning multidimensional signal processing. 1998. Linköping Studies in Science and Technology, Dissertations, No. 531, Department of Electrical Engineering, Linköping University, Linköping, Sweden.

[17] H. E. Cetingul and R. Vidal. Intrinsic mean shift for clustering on stiefel and grassmann manifolds. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[18] J.-M. Cheng, J. R. Beveridge, B. Draper, M. Kirby, H. Kley, and C. Peterson. Illumination face spaces are idiosyncratic. In *The International Conference on Image Processing, Computer Vision, and Pattern Recognition*, pages 390–396, 2006.

[19] J.-M. Cheng, M. Kirby, H. Kley, C. Peterson, B. Draper, and J. R. Beveridge. Recognition of digital images of the human face at ultra low resolution via illumination spaces. In *Asian Conference on Computer Vision*, pages 733–743, 2007.

[20] Y. Cheng. Mean shift, mode seeking, and clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17:790–799, 1995.

[21] Y. Chikuse. Statistics on special manifolds. lecture notes in statistics. 2003. Springer, New York.

[22] T.-J. Chin and D. Suter. A new distance criterion for face recognition using image sets. In *Asian Conference on Computer Vision*, pages 549–558, 2006.

[23] J. Conway, R. Hardin, and N. Sloane. Packing lines, planes, etc.: Packings in grassmannian spaces. *Experimental Mathematics*, 5(2):139–159, 1996.

[24] F. De la Torre, R. Gross, S. Baker, and V. Kumar. Representational oriented component analysis (roca) for face recognition with one sample image per training class. In *IEEE International Conference on Computer Vision and Pattern Recognition, San Diego*, 2005.

[25] L. De Lathauwer. Signal processing based on multilinear algebra. 1997. Ph.D. Thesis, K.U. Leuven, E.E. Dept.-ESAT, Belgium.

[26] L. De Lathauwer. A survey of tensor methods. In *IEEE International Symposium on Circuits and Systems, Taiwan*, 2009.

[27] L. De Lathauwer, B. D. Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM J. Matrix Anal. Appl.*, 21(4):1253–1278, 2000.

[28] A. Edelman, R. Arias, and S. Smith. The geometry of algorithms with orthogonal constraints. *SIAM J. Matrix Anal. Appl.*, (2):303–353, 1999.

[29] W. Fan and D.-Y. Yeung. Locally linear models on face appearance manifolds with application to dual-subspace based classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[30] D. Fiori. Formulation and integration of learning differential equations on the stiefel manifold. *IEEE Transactions on Neural Networks*, (6):1697–1701, 2005.

[31] A. Fitzgibbon and A. Zisserman. Joint manifold distance: a new approach to appearance based clustering. In *IEEE Conference on Computer Vision and Pattern Recognition, Wisconsin*, 2003.

[32] Y. Freund and R. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

[33] Y. Fu and T. Huang. Image classification using correlation tensor analysis. *IEEE Transactions on Image Processing*, (2):226–234, 2008.

[34] K. Fukui, O.Yamaguchi, and K. K.Suzuki. Face recognition under variable lighting condition with constrained mutual subspace method. *Trans. IEICE (DII) (in Japanese)*, 82(4):613–620, 1999.

[35] K. Fukui, B. Stenger, and O. Yamaguchi. A framework for 3d object recognition using the kernel constrained mutual subspace method. In *Asian Conference on Computer Vision*, pages 315–324, 2006.

[36] K. Fukui and O. Yamaguchi. Face recognition using multi-viewpoint patterns for robot vision. In *Robotics Research, The Eleventh International Symposium, ISRR*, pages 192–201, 2005.

[37] K. Fukui and O. Yamaguchi. The kernel orthogonal mutual subspace method and its application to 3d object recognition. In *Asian Conference on Computer Vision*, pages 467–476, 2007.

[38] A. Georghiades, P. Belhumeur, and D. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):643–660, 2001.

[39] J. Gower and G. B. Dijksterhuis. Procrustes problems. 2004. Oxford University Press.

[40] J. Hamm and D. Lee. Extended grassmann kernels for subspace-based learning. In *Advances in Neural Information Processing System*, pages 601–608, 2008.

[41] J. Hamm and D. Lee. Grassmann discriminant analysis: a unifying view on subspace-based learning. In *International Conference on Machine Learning*, pages 376–383, 2008.

[42] X. He, D. Cai, and P. Niyogi. Tensor subspace analysis. In *NIPS*, 2005.

[43] H. Hotelling. Relations between two sets of variants. *Biometrika*, 28:321–377, 1936.

[44] K. Jia and D.-Y. Yeung. Human action recognition using local spatio-temporal discriminant embedding. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[45] X. Jiang, Y. Kong, J. Huang, R. Zhao, and Y. Zhang. Learning from real images to model lighting variations for face images. In *European Conference on Computer Vision, Marseille, France*, 2008.

[46] T. Kanade. Personal correspondence.

[47] T.-K. Kim, O. Arandjelović, and R. Cipolla. Boosted manifold principal angles for image set-based recognition. *Pattern Recognition*, 40(9):2475–2484, 2007.

[48] T.-K. Kim and R. Cipolla. Canonical correlation analysis of video volume tensors for action categorization and detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(8):1415–1428, 2009.

[49] T.-K. Kim, J. Kittler, and R. Cipolla. Discriminative learning and recognition of image set classes using canonical correlations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):1–14, 2007.

[50] T.-K. Kim, S.-F. Wong, and R. Cipolla. Tensor canonical correlation analysis for action classification. In *IEEE Conference on Computer Vision and Pattern Recognition, Minnesota*, 2007.

[51] M. Kirby. *Geometric Data Analysis*. John Wiley and Sons, Inc, 2001.

[52] T. Kohonen and E. Oja. Fast adaptive formation of orthogonalizing filters and associative memory in recurrent networks of neuron-like elements. *Biological Cybernetics*, 21(2):85–95, 1976.

[53] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3), September 2009.

[54] S. Kozlov. Geometry of real grassmann manifolds. parts i, ii. *Journal of Mathematical Sciences*, 100(3):2239–2253, 2000.

[55] M. Lades, J. Vorbruggen, J. Buhmann, J. Lange, C. von der Malsburg, R. Wurtz, and W. Konen. Distortion invariant object recognition in the dynamic link architecture. *IEEE Transactions on Computers*, 42(3):300–311, March 1993.

[56] L. Lam and C.-Y. Suen. Application of majority voting to pattern recognition: An analysis of its behavior and performance. *IEEE Transactions on Systems, Man and Cybernetics, Part A*, 27(5):553–568, Sept. 1997.

[57] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfield. Learning realistic human actions from movies. In *IEEE Conference on Computer Vision and Pattern Recognition, Alaska*, 2008.

[58] J. Lee. Introduction to smooth manifolds. 2003. Springer.

[59] K. Lee, J. Ho, and D. Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27:684–698, 2005.

[60] P. Y. Lee. Geometric optimization for computer vision, 2005.

[61] X. Li, K. Fukui, and N. Zheng. Boosting constrained mutual subspace method for robust image-set based object recognition. In *IJCAI*, 2009.

[62] X. Li, K. Fukui, and N. Zheng. Image-set based face recognition using boosted global and local principal angles. In *Asian Conference on Computer Vision*, 2009.

[63] X. Li, W. Hu, Z. Zhang, X. Zhang, and G. Luo. Robust visual tracking based on incremental tensor subspace learning. In *IEEE International Conference on Computer Vision*, 2007.

[64] D. Lin, S. Yan, and X. Tang. Pursuing informative projection on grassmann manifold. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 386–389, 2005.

[65] J. Liu, S. Chen, Z.-H. Zhou, and X. Tan. Single image subspace for face recognition. In *IEEE International Workshop on Analysis and Modeling of Faces and Gestures*, pages 205–219, oct 2007.

[66] X. Liu, A. Srivastava, and K. Gallivan. Optimal linear representations of images for object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(5):662–666, 2004.

[67] Y. M. Lui, J. R. Beveridge, B. A. Draper, and M. Kirby. Image-set matching using a geodesic distance and cohort normalization. In *IEEE International Conference on Automatic Face and Gesture Recognition, Amsterdam, The Netherlands*, 2008.

[68] Y. Ma, J. Košecká, and S. Sastry. Optimal motion from image sequences: A riemannian viewpoint, 1998. Technical Report No. UCB/ERL M98/37, EECS Department, University of California, Berkeley.

[69] K.-I. Maeda, O. Yamaguchi, and K. Fukui. Towards 3-dimensional pattern recognition. In *Structural, Syntactic, and Statistical Pattern Recognition, Joint IAPR International Workshops*, pages 1061–1068, 2004.

[70] K.-I. Maeda, O. Yamaguchi, and K. Fukui. A fundamental discussion of 3-dimensional pattern matching using canonical angles between subspaces for the purpose of differentiating a face and its photograph. *Systems and Computers in Japan*, (9):11–20, 2007.

[71] J. Manton. Optimization algorithms exploiting unitary constraints. *IEEE Transactions on Signal Processing*, 50(3):635–650, 2002.

[72] Q. McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, pages 153–157, 1947.

[73] T. Melzer, M. Reiter, and H. Bischof. Appearance models based on kernel canonical correlation analysis. *Pattern Recognition*, 36:1961–1971, 2003.

[74] J. Mendel. Tutorial on higher-order statistics (spectra) in signal processing and system theory: theoretical results and some applications. *Proceedings of the IEEE*, 79(3):278–305, 1991.

[75] T. Moeslund and E. Granum. A survey of computer vision based human motion capture. *Computer Vision and Image Understanding*, 81:231–268, 2001.

[76] Y. Moses, Y. Adini, and S. Ullman. Face recognition: The problem of compensating for changes in illumination direction. In *European Conference on Computer Vision, Marseille*, pages 286–296, 1994.

[77] M. Nakayama and T. Kumakura. Face identification performance using facial expressions as perturbation. In *ICANN*, 2005.

[78] J. Niebles, H. Wang, and L. Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. *International Journal of Computer Vision*, 79(3):299–318, 2008.

[79] M. Nishiyama, O. Yamaguchi, and K. Fukui. Face recognition with the multiple constrained mutual subspace method. In *AVBPA*, pages 71–80, 2005.

[80] M. Nishiyama, M. Yuasa, T. Shibata, and T. Wakasugi. Recognizing faces of moving people by hierarchical image-set matching. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[81] E. Oja and J. Parkkinen. On subspace clustering. In *IEEE Conference on Pattern Recognition*, pages 692–695, 1984.

[82] D.-S. Pham and S. Venkatesh. Robust learning of discriminative projection for multicategory classification on the stiefel manifold. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[83] P. Phillips, H. Moon, S. Rizvi, and P. Rauss. The feret evaluation methodology for face recognition algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10):1090–1104, 2000.

[84] R. Poppe. A survey on vision-based human action recognition. *Image and Vision Computing (appear)*, 2010.

[85] L. Qi, W. Sun, and Y. Wang. Numerical multilinear algebra and its applications. *Frontiers of Mathematics in China*, 2(4):501–526, October 2007.

[86] L. Qiu, Y. Zhang, and C.-K. Li. Unitarily invariant metrics on the grassmann space. *SIAM Journal on Matrix Analysis and Applications*, 27(2):507–531, 2005.

[87] R. Ramamoorthi and P. Hanrahan. On the relationship between radiance and irradiance: determining the illumination from images of a convex lambertian object. *Journal of the Optical Society of America A*, 18(10):2448–2459, 2001.

[88] S. Rana, W. Liu, M. Lazarescu, and S. Venkatsh. Recognising faces in unseen modes: A tensor based approach. In *IEEE International Conference on Computer Vision and Pattern Recognition*, 2008.

[89] S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, pages 2323–2326, 2000.

[90] P. Saisan, G. Doretto, Y.-N. Wu, and S. Soatto. Dynamic texture recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2001.

[91] C. Schüldt, I. Laptev, and B. Caputo. Recognizing human actions: A local svm approach. In *International Conference on Pattern Recognition, Cambridge, UK*, 2004.

[92] H. Seung and D. Lee. The manifold ways of perception. *Science*, pages 2268–2269, 2000.

[93] S. Shan, W. Zhang, Y. Su, X. Chen, and W. Gao. Ensemble of piecewise fda based on spatial histograms of local (gabor) binary patterns for face recognition. In *International Conference on Pattern Recognition, Hong Kong*, pages 606–609, 2006.

[94] T. Sim, S. Baker, and M. Bsat. The cmu pose, illumination, eand expression (pie) database. In *IEEE International Conference on Automatic Face and Gesture Recognition*, 2002.

[95] T. Sim and T. Kanade. Combining models and examplars for face recognition: An illuminating example. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2001.

[96] P. Simard, Y. LeCun, and J. Denker. Efficient pattern recognition using a new transformation distance. In *Advances in Neural Information Processing System 5*, pages 50–58, 1992.

[97] A. Simon. Calculus with analytic geometry. 1982. Scott, Foresman and Company.

[98] A. Srivastava and X. Liu. Tools for application-driven linear dimension reduction. pages 136–160, 2005.

[99] R. Subbarao and P. Meer. "nonlinear mean shift for clustering over analytic manifolds". In *IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[100] R. Subbarao and P. Meer. "subspace estimation using projection based m-estimators over grassmann manifolds". In *European Conference on Computer Vision*, 2006.

[101] X. Sun, L. Wang, and J. Fe. Further results on the subspace distance. *Pattern Recognition*, 40:328–329, 2007.

[102] X. Tan, S. Chen, Z.-H. Zhou, and F. Zhang. Face recognition from a single image per person : A survey. *Pattern Recognition*, pages 1725–1745, 2006.

[103] X. Tan and B. Triggs. Fusing gabor and lbp feature sets for kernel-based face recognition. In *IEEE International Workshop on Analysis and Modeling of Faces and Gestures*, pages 235–249, oct 2007.

[104] J. Tenenbaum, V. de Silva, and J. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, pages 2319–2323, 2000.

[105] P. Turaga and R. Chellappa. Locally time-invariant models of human activities using trajectories on the grassmannian. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[106] P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea. Machine recognition of human activities: A survey. *IEEE Transactions on Circuits and Systems for Video Technology*, (11):1473–1488, 2008.

[107] P. Turaga, A. Veeraraghavan, and R. Chellappa. Statistical analysis on stiefel and grassmann manifolds with applications in computer vision. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[108] N. Vasconcelos and A. Lippman. A multiresolution manifold distance for invariant image similarity. *IEEE Transactions on Multimedia*, pages 127–142, 2005.

[109] M. A. O. Vasilescu and D. Terzopoulos. Multilinear image analysis for facial recognition. In *International Conference on Pattern Recognition, Quebec City, Canada*, pages 511–514, 2002.

[110] L. Wang, X. Wang, and J. Fe. Subspace distance analysis with application to adaptive bayesian algorithm for face recognition. *Pattern Recognition*, 39:456–466, 2006.

[111] R. Wang, S. Shan, X. Chen, and W. Gao. Manifold-manifold distance with application to face recognition based on image set. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[112] T. Wang, A. G. Backhouse, and I. Y. Gu. Online subspace learning on grassmann manifold for moving object tracking in video. In *IEEE International on Acoustics, Speech and Signal Processing, Las Vegas*, pages 969–972, April 2008.

[113] T. Wang and P. Shi. Kernel grassmannian distances and discriminant analysis for face recognition from image sets. *Pattern Recognition Letters*, 30(13):1161–1165, 2009.

[114] L. Wiskott, J.-M. Fellous, N. Kruger, and C. Malsburg. Face recognition by elastic bunch graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:775–779, 1997.

[115] L. Wolf and A. Shashua. Learning over sets using kernel principal angles. *Journal of Machine Learning Research*, 4:913–931, 2003.

[116] S.-F. Wong and R. Cipolla. Extracting spatiotemporal interest points using global information. In *IEEE International Conference on Computer Vision*, 2007.

[117] Y.-C. Wong. Differential geometry of grassmann manifolds. *Proc. Nat. Acad. Sci.*, 47:589–594, 1967.

[118] O. Yamaguchi, K. Fukui, and K. Maeda. Face recognition using temporal image sequence. In *International Conference on Face and Gesture Recognition, Nara, Japan*, pages 318–323, 1998.

[119] B. Zhang, S. Shan, X. Chen, and W. Gao. Histogram of gabor phase patterns (hgpp) : A novel object representation approach for face recognition. *IEEE Transactions on Image Processing*, 16:57–68, 2007.

[120] L. Zhang and D. Samaras. Face recognition under variable lighting using harmonic image examplar. In *IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN*, 2003.

[121] L. Zhang and D. Samaras. Face recognition from a single training image under arbitrary unknown lighting using spherical harmonics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28:351–363, 2006.

[122] S. Zhou and R. Chellappa. Beyond one still image: Face recognition from multiple still image or video sequence. pages 547–576, 2006. in Face Processing Advance Modeling and Methods, Edited by W. Zhao and R. Chellappa, Elsevier.

[123] B. Zitova and J. Flusser. Image registration methods: A survey. *Image and Vision Computing*, 21:977–1000, 2003.

[124] J. Zou, Q. Ji, and G. Nagy. A comparative study of local matching approach for face recognition. *IEEE Transactions on Image Processing*, 16:2617–2628, 2007.

[125] G. Zuccon, L. Azzopardi, and C. J. van Rijsbergen. Semantic spaces: Measuring the distance between different subspaces. In P. Bruza, D. Sofge, W. F. Lawless, K. van Rijsbergen, and M. Klusch, editors, *Quantum Interaction*, volume 5494 of *Lecture Notes in Computer Science*, pages 225–236. Springer, 2009.