Computations on the Grassmann Manifold and Affine-Permutation Invariance in Shape Representation

A Dissertation

Presented to the Faculty of the Graduate School

of Carnegie Mellon University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philoshophy

by

David Sepiashvili

dasepi@qanalyst.org

May 12, 2006

Carnegie Mellon University

Electrical and Computer Engineering

5000 Forbes Avenue, Pittsburgh, PA

Dedicated to my parents

ACKNOWLEDGEMENTS

I wish to acknowledge and thank those people who contributed to this thesis.

I owe my deepest gratitude to my advisor, Dr. José M. F. Moura for many insightful conversations during the development of the ideas in this thesis, and for helpful comments on the text. Throughout my thesis-writing period, he provided encouragement, support, sound advice, and good teaching.

I would like to express my gratitude to my committee, Drs. João Xavier, Tsuhan Chen, Jelena Kovacevic, Irene Fonseca. It is difficult to overstate my gratitude to João Xavier for his enthusiasm and great efforts to discuss manifolds and make them clear and simple, and also for a bunch of ideas in other approaches. I am especially thankful to Tsuhan Chen for fruitful conversations on related to the Structure from Motion problems. I would like to thank Jelena Kovacevic, Irene Fonseca who made many helpful suggestions. Many thanks and much respect to Victor Ha and Franz Franchetti for feedback on research ideas and practice presentations. This work would not be complete without continuous moral support from my friends Hy Goldstoff, Elithabeth Stirr, Aca Gacic, Iva Miljkovic, Rick Butts, Dana Ferrari, Constantin Kostenko, Crysta Lynn, and many more. I thank my officemates, and hopefully still friends, for putting up with me during my dissertation period. Also, thanks to the Electrical Engineering Department for support and assistance throughout these years, especially to Carol Patterson, Elaine Lawrence and Lynn Philibin. I would also like to thank all the rest of the academic and support staff of the CMU.

Finally, I am forever indebted to my parents, Lana and Al, for their love, understanding, endless patience and encouragement when it was most required. I cannot thank them enough for all the opportunities they have given me in my life.

ABSTRACT

We present a novel approach to computations on the Grassmann manifold and define a locally Riemmanian affine-permutation invariant shape space for the task of shape representation. The definition of a measure of similarity between shapes cannot be based on the geometry of Euclidean space. One of the essential concepts here is defining a shape to be a set of all equivalent representations obtained by acting on a rigid object with a class of distortions and then mapping each shape as a point of a geometric manifold. The underlying metric of the non-Euclidean space intrinsically defines an unlabeled-view-independent similarity measure between the shapes. We consider affine distortions and distortions induced by permutation of the unlabeled feature points. We solve this challenging problem of finding a similarity measure and of finding a locally geometric manifold with an unlabeled-view-independent metric properly defined on it. We construct geometric shape spaces for 2D and 3D rigid objects and define on these spaces distances among shapes that are invariant to a class of distortions — affine transformations and permutations of unlabeled feature points. We define equivalency sets of configurations of p-dim objects. Each equivalency class collects the configurations of a p-dim object that are transformable into each other under the class of affine transformations and the class of permutations of the feature points.

We give an analytical mapping to the neighborhood of the real Grassmann manifold as a solution to the distance minimization problem. As the true distance minimization is not easily solvable, we present an approximate solution using linear programming and quadratic programming.

The main result is the development of a new modeling and computational framework, and related general-purpose calculation formulas and algorithms. We derive explicit analytical linear solutions for the exponential map and the logarithmic map. We give a linear and efficient algorithm for the computation of the approximate barycentric combinations of points on the Grassmann manifold. The derived original general-purpose explicit formulas for the real Grassmann manifold allow easy computation of the basic characteristics, such as measuring distances between points (shapes) and computing sample means and higher order statistics, for the task of shape representation. They can also be used in many other fields besides shape representation theory. In the thesis, these concepts are directly applied to the affine-invariant shape space, allowing computation of similarity between shapes as well as the sample mean and higher order moments of a constellation of shapes. This work is of great practical importance as it provides with efficient and robust ways of applying the theory to real world applications, such as multi-author handwritten digit recognition. Our broad contribution adds another dimension to the problem of representation of a geometric object and bridges a gap between image processing algorithms in Riemannian spaces and conventional algorithms embedded for simplicity in Euclidean spaces.

Keywords: shape theory, shape representation, affine-permutation group, unlabeled-view-independent similarity measure, distortion invariant metric, geometric invariance, affine transformations, permutation distortions, correspondence problem, feature matching, computer vision, Riemannian geometry, equivalency classes, Grassmann manifold, geometric algorithms, logarithmic map, barycentric combinations, statistical inference, sample mean.

CONTENTS

LIST OF FIGURES

1. INTRODUCTION

1.1 Organization

The dissertation studies the problems of computations on the Grassmann manifold and affine-permutation invariance in shape representation. It is presented as a progression of topics.

In Chapter 1, we start with the problem definition and motivation. Also we consider briefly the history of shape theories and some prior work. The survey and critical assessment of shape theories help us to establish background on the underlying theory and to formulate key concepts on the representation of geometric objects under different viewing conditions. One of the essential concepts is defining a shape to be a set of all equivalent representations obtained by acting on a rigid object with a class of distortions and then mapping each shape as a point of a geometric manifold. The underlying metric of the non-Euclidean space intrinsically defines an unlabeled-view-independent similarity measure between the shapes. We consider affine distortions and distortions induced by permutation of the unlabeled feature points. This goes well beyond Kendall's shape theory, see [65], that is restricted to motions with uniform scaling with shapes being defined by labeled feature points.

Next, we describe the contributions of the dissertation. The main result is the development of a new modeling and computational framework, and related general-purpose calculation formulas and algorithms. Our broad contribution adds another dimension to the problem of representation of a geometric object. We construct geometric shape spaces with distances and local geometry defined in such a way that they achieve invariance to the following class of modeled distortions: simultaneous affine transformations and permutations of unlabeled feature points.

Chapter 2 addresses shape representation. We define shape and derive a mapping that allows to represent each shape as a point on a Riemannian manifold. We start by providing a brief overview of image representation and introduce the reader to the concepts of affine space and its symmetry group, and relate these concepts to the affine distortions of a rigid object. We also discuss here other important issues, such as permutations of the feature points of the object. These concepts lead us to definitions of shape views and shape configuration matrices. Next, we define the affine-permutation group and give its decomposition into three simpler groups. We give definitions of group actions and quotient spaces formed by group actions. Finally, we introduce an injective mapping from an abstract space of shapes to a Riemannian space. In addition, we summarize the important algorithm building blocks one needs to address in order to be able to perform various algorithms on Riemannian spaces.

In Chapter 3, we consider embedded Riemannian manifolds. We describe key notions from the theory of manifolds in a very informal way. We explain the notions of vector fields and flows, geodesics and associated geodesic vector fields, and the notion of intrinsic distance. Also, we give a more intuitive representation for the exponential map as an addition of a vector to a point, and its inverse — the logarithmic map — as a subtraction of two points. In addition we give various definitions for the mean on a Riemannian manifold and define a notion of barycentric combinations. Finally, we apply these notions to the example of the orthogonal group $\mathcal{O}_{\mathrm{n}}$ when it is considered as an embedded Riemannian manifold.

Chapter 4 is concerned with computations on the real Grassmann manifold $\mathcal{G}r_{\mathrm{n,p}}$. It gives a representation for the affine-invariant shape space. We start by defining the Grassmann manifold and discuss various numerical representations of its points. We also give the decomposition of the Grassmann manifold as a quotient of the orthogonal group. Then we give explicit analytical linear solutions for the exponential map and the logarithmic map. We give a linear and efficient algorithm for the computation of the approximate barycentric combinations of points on the Grassmann manifold and discuss related notions of principal

angles, cut locus, and maximal radii for normal regular balls that allow to establish conditions for uniqueness and existence of the barycentric combination. The general-purpose explicit formulas that we derive for the real Grassmann manifold allow easy computation of basic quantities, such as measuring distances between points and computing sample means and higher order statistics. These are important on such tasks as object recognition and categorization. They can be used in many other theories besides shape representation theory. However, in particular, when affine-invariant shapes are considered as points on the Grassmann manifold, these concepts are directly applied to the affine-invariant shape space, allowing computation of similarity between shapes as well as a sample mean and higher order moments of a constellation of shapes.

Chapter 5 goes a step forward and establishes a representation for the affine-permutation-invariant shape space. In a way, this is a folded Grassmann manifold $\mathcal{G}r_{\mathrm{n,p}}/\mathcal{P}_{\mathrm{n}}$ under the group action of the finite permutation group $\mathcal{P}_{\mathrm{n}}$. This quotient space is not a Riemannian manifold. However, we establish in Chapter 5 a conjecture that the affine-permutation-invariant shape space locally looks like the neighborhood of the real Grassmann manifold $\mathcal{G}r_{\mathrm{n,p}}$. We give an analytical mapping that maps neighborhoods in the shape space to the neighborhoods of the real Grassmann manifold. This map is the solution to the distance minimization problem. As the true distance minimization is not easily solvable, we present an approximate solution using linear programming and quadratic programming. In addition, based on our experiments, we establish another conjecture. It states that if the minimized distance found as a result of linear programming is large, then the true minimized distance would have been large as well. In other words, it tells us that the linear programming solution is a mapping that preserves distances locally. For this reason, given two shapes, we can establish how similar the shapes are, and if they are sufficiently similar we can compute their distance as well as perform other geometric concepts derived for the Grassmann manifold. In particular, we can compute a sample mean and higher order moments of a constellation of shapes that are sufficiently similar to their sample mean.

In Chapter 6, we present applications of our innovative approach, the basic concepts and

derived formulas, to the classification problem on shape spaces. We show how to compute the sample mean for a constellation of shapes. We give an example of shape morphing by means of connecting two shapes with a minimal geodesic segment and following it. We also give an example of shape identification when applied to multi-author handwritten digit recognition. These applications show practical importance of our work. We finally discuss ways of defining data-driven sub-manifolds of the shape space.

Finally, in Chapter 7 we summarize the results of our work.

1.2 Problem definition and motivation

The representation of a geometric object is one of the most challenging problems in computer vision and machine learning. Learning the similarity among objects is one of its key concepts. If we do not have effective similarity measures, the important tasks of object recognition and categorization can not be performed reliably. The problem is a difficult one as we may find appearances of similar objects to be very different from each other under different viewing conditions. Moreover, different views of the same object will not have their features in correspondence with each other.

Defining an unlabeled-view-independent similarity measure is a combinatorial computationally expensive problem. Another difficulty one faces is that the similarity measure should not be an Euclidean one, therefore numerical algorithms to compute distances cannot use the geometry of the Euclidean space. Thus, we need to solve an even more challenging problem that of finding not just a similarity measure, but a geometric manifold with an unlabeled-view-independent metric properly defined on it.

The main problem of computer vision is studying how do the captured images encode properties of the world, such as illumination and material, texture, shape, and spatial motion of the objects. In this research, we are concerned only with the representation of object shapes and assume that segmentation, compensation for the illumination changes, texture, and motion recovery are already solved or need not be solved for a specific setup. In this thesis, we give a new innovative approach toward the solution of this problem, which allows

to reliably represent object shapes and minimize (reduce) the combined affine-permutation shape distortion.

We will consider both 3D and 2D shapes of rigid objects and the construction of geometric shape spaces with distances defined in such a way that they achieve invariance to a specific class of modeled distortions. The distortions we study are simultaneously applied affine transformations and permutations of unlabeled feature points.

The geometric approach to signal processing and image processing is not common, except for few exceptions [110, 100]. The goal of this research is to bridge a gap between image processing algorithms in the Riemannian spaces and conventional algorithms embedded for simplicity in Euclidean spaces, and to take advantage of looking at these algorithms from this new perspective. A good review of numerous estimation problems in the Riemannian spaces is given in [102].

The thesis applies our similarity measures to various examples, including an example of multi-author handwritten digit recognition.

1.3 Prior work: survey and critical assessment of shape theories

There are a number of different shape theories. Usually, humans define a *"shape"* to be the appearance of an object that is invariant under Euclidean similarity transformations. In the computer vision community, imaging sensors are frequently modeled by a pinhole perspective projection camera model, which is frequently approximated by an affine camera model, [37]. Thus, an Euclidean similarity transformation of a rigid object is approximated by an affine transformation of its views. In other words, an affine distortion of object views is an approximation to changes between different views of the same object undergoing an Euclidean similarity transformation, [37]. Then a shape of an object is a collection of its views undergoing affine transformations. In order to be able to perform various tasks on shapes, such as measuring similarity between shapes, computing sample means and putting probability distributions on a space of shapes, one needs to achieve invariance to affine distortions in the space of object views (captured images).

There are three major approaches to treating affine invariance in the space of object views: estimation of an alignment transformation, computation of invariants, and definition of affine-invariant similarity measure, or even geometry.

The first approach measures similarity between shapes by first solving for correspondences between points on the two views (images), and then using the correspondences to estimate an aligning transformation. Then one uses some sort of similarity measure between the two aligned views, such as mean square error. Typical examples of such approach are given in [12, 58, 85]. An overview of image alignment techniques is given in [9]. Solving for correspondences is also known as a feature matching problem, see [73, 72]. In the computer vision community, there are attempts to bring Gestalt laws into practice. One of the first approaches into this direction is given in [97]. The authors find an innovative way to solve the visual correspondence problem by using the Gestalt law of proximity, and creating an algorithm for associating the features of two patterns. However with these lines of approaches, which are typically considered in the computer vision and signal processing communities, one cannot define a geometry on the space of shapes. Thus, one can only hope for an efficient estimation of the transformation and a good similarity measure between the shapes.

The second approach is based on computing different functions that are invariant to the relevant group of transformations, such as affine transformations. As an example, one could consider a computation of central moments. A detailed treatment on invariants for model-based recognition is given in [82]. The main shortcoming of the approach is that one needs to have sufficiently large number of invariants computed in order to be able to recover a good approximation to the original object view (image). Otherwise, an obtained similarity measure will not discriminate well between different shapes. However, it is practically hard to obtain a large number of invariants. For example, consider computing central moments. This computation is numerically very sensitive to noise, making it essentially impossible to calculate moments of sufficiently high orders. Besides, very often it is difficult to define a meaningful distance between shapes based on their invariants. Another shortcoming is that

the space of invariants does not allow to perform geometric operations, such as an averaging of a constellation of objects. This approach is mainly of theoretical interest. In practice, it does not work well.

Finally, the third approach defines an affine-invariant similarity measure on factor spaces of object views. In this approach one would simply form a true shape space with both distance and geometry defined on it. A typical strategy is to start by defining a finite space of shape configuration matrices, or an infinite dimensional space of configuration atlases. Then one defines a group action on this space by acting on it with the group of distortions under the consideration. Group actions partition the space, forming equivalency classes. Under this approach, one is looking not just for an affine-invariant similarity measure, but for a geometric space that will allow to perform various tasks on shapes, such as measuring similarity between shapes, computing sample means and putting probability distributions on a space of shapes. Now let us take a look into a history of various shape theories that fall under this approach. There are two major classes. Once equivalency classes are formed, one has a choice to either search for a canonical member of the equivalency class, or to work within the space of equivalency classes.

One popular form of the first class, where one is looking for a canonical member of each equivalency class, is whitening, in which an affine transformation is applied to a set of points such that they have zero average and the identity covariance matrix. The disadvantage of the method is that there are still remaining degrees of freedom left. Another form of normalization, obtained by bringing pivot points in the configuration to a standard location suffers from being arbitrary and thus highly sensitive to noise. The work given in [45] falls under this approach. Related to the approach are also works that define medial or principal axes of an object.

In the second class, one of the first sound approaches to the problem seems to have been motivated by Kendall's 1977 paper [64]. Kendall introduces the term *"shape space"* assuming that the shape of an object is captured by the shape of a finite subset of its points. In his shape theory, he argues that it is not possible to measure the similarity between

possible shapes of an object in standard Euclidean coordinates. Thus, he is looking for new shape spaces and their topological and geometric properties. Dryden and Mardia in their book on shape theory [32], following Kendall's intuitive definition of a shape, define a *shape to be "all the geometrical information that remains when location, scale, and rotational effects are filtered out from an object."* That is, different shape spaces are obtained by factoring out a class of allowed transformations, making shapes in these spaces invariant to the modeled transformations.

Various shape spaces are also discussed in [14], but, unfortunately, the author is assuming a labeled set of feature points, and is primarily concerned with finding statistical density of a shape. He is not studying the underlying geometric structure of the shape spaces.

Another very common shape theory is given by Grenander and his collaborators [43, 60, 44]. These authors describe shapes in terms of closed curves that define their boundaries and outlines. The geometry of the space of curves is studied in [111], also see [77].

Yet another approach, e.g., in [22], is to represent simple algebraic objects, such as cylinders, by Lie algebras, and then model more complex objects as a collection of patches of these simpler algebraic objects. Under this approach, each shape of an object is a collection of invariants of Lie algebras that achieve invariance to Euclidean (or affine) transformation. Unfortunately, a concept of a shape space is not applicable here.

In this research, we adopt the concept of a shape given by Kendall [65]. However, our research is different in a number of ways. Kendall assumes a *labeled* finite subset of the object's feature points. In our case, the feature points are *not* labeled, causing permutation distortions of the feature points. Also, the class of allowed transformations in Kendall's work and in our work is quite different. Kendall restricts the class of motions to rigid motions and uniform scaling. In our research, we are modeling a class of affine motions. A concept of the general shape space with labeled feature points and affine invariance is not new. It goes back to Ambartzumian, [6], and is also discussed in [32]. However, Ambartzumian's work is primarily concerned with the theory of invariant measures, stochastic point processes, and their distributions in the shape spaces. It is not easily applicable to practical applications.

Our work bridges the gap between the theoretical part of shapes space and a practical efficient implementation of it by providing with computational framework.

This research contains the work in [98], see also [45, 48, 47, 46]. It is worth to note that work [11] follows our framework, presented for the first time in [98].

1.4 Contributions of the dissertation

The dissertation makes the following substantive original contributions:

1. Bridging a gap between image processing algorithms in the Riemannian spaces and conventional algorithms embedded for simplicity in Euclidean spaces, and taking advantage of looking at these algorithms from this new perspective;

2. Introducing noise and dimensionality (number of sample points) resilient efficient shape representation by left cosets (equivalency classes) that avoids the nontrivial task of recovering canonical members for each equivalency class;

3. Deriving efficient linear equations, general-purpose expressions and formulas for writing numeric algorithms on the Grassmann manifold — a non-Euclidean space of curvature, including explicit and efficient linear formulas for computing the logarithmic map and the Riemannian mean;

4. Defining a novel approach for simultaneously achieving invariance to affine transformations of a rigid object and solving for the feature correspondence problem;

5. Finding not just a similarity measure between shapes but defining a locally geometric shape space with unlabeled-view-independent metric on it, allowing not only to measure distance between different shapes, but also to perform other geometry operations such as finding a sample mean or following geodesics;

6. Showing the practical importance of the findings for various computer vision and image processing applications and presenting real world case studies, including examples of classification and statistical inference on the shape space.

2. FROM SHAPES TO MANIFOLDS AND SHAPE REPRESENTATION

2.1 Preliminaries

How should one represent geometric objects numerically? In physics, it is common practice to consider objects to be sets of particles with some special properties, such as mass, so one could say that objects live in a space consisting of particles. But how do we model the space of particles? How do we represent them in a digital computer?

In the view of Newtonian Mechanics, a particle is modeled as a point, and the position of this point is determined with respect to a fixed coordinate system (*"frame"*) by coordinates of a vector connecting the origin of the frame with the point with respect to the chosen basis. But is the assumption of a known origin always valid?

We live in a world of 3D objects. Very frequently, we capture images of the world using 2D and 3D sensors that are arbitrarily located and oriented with respect to the world. As the relative 3D position of the object is unknown and often changes in time, the view of the object captured by the sensor looks distorted in different images. The distortions are due to the fact that the coordinate systems of the sensors do not coincide with the coordinate system of the world. Let us denote dimensionality of the space under consideration by p (e.g., $p = 2$ or $p = 3$). Then, the space of points with a fixed frame is isomorphic to a vector space $\mathbb{R}^p$. As we can see, the point corresponding to the zero vector (the origin) plays a special role, when there is really no reason to have a privileged origin.

Inspired by this fact, one would like to define a space that has no special origin pre-selected. Such space is called an *"affine space."* Informally speaking, an affine space is a vector space that has forgotten its origin. Throughout the dissertation the particles are

described in this space.

Another important concept that will be addressed here is the concept of symmetries of a geometric object, which form a group, called the *"symmetry group."* The symmetry group of a geometric object is, intuitively, the set of all transformations that can be applied to the object without changing it. These transformations are non-degenerate maps that preserve the properties of the underlying space, the affine space in our case.

The key points that will be addressed in this chapter are:

- ***motivation for considering affine transformations — Sections 2.2-2.3***
- ***definitions of the affine space and its symmetry group — Section 2.4***
- ***shapes, shape views and configuration matrices — Sections 2.5-2.7***
- ***affine-permutation group and its decomposition — Section 2.8***
- ***shape space reduction as a quotient of manifolds — Section 2.9***
- ***working with shapes as points of a Riemannian manifold — Section 2.10***

We start by motivating the reader further as to why we should be working in the affine space. Then we define an affine space in a group theoretic framework and present preliminaries on working in the quotient space that results after a group of symmetries has been factored out. Then we give formal definitions of a shape, a shape view, and a shape configuration matrix. We also derive the affine-permutation group as a group of distortions that acts on a rigid object and give its decomposition into simpler groups. We investigate shape space representation as a quotient of manifolds under the action of these groups. We also provide the reader with a step-by-step mapping from an abstract space of shapes to a Riemannian space, shown in Figure 2.2. In addition, we summarize the important algorithm building blocks one needs to address in order to be able to perform various algorithms in the Riemannian spaces. These preliminaries and background is needed in the subsequent chapters of the thesis.

Now we motivate the use of 3D affine transformations.

2.2 Why consider 3D affine transformation for 3D shapes?

Frequently, imaging sensors are modeled as a pinhole perspective projection of the 3D world onto the 2D image plane. A coarser but very popular approximation is an affine projection camera model. In this model, the observed objects are assumed to undergo an affine motion before being projected ortographically onto the image plane.

Now suppose that a set of points of interest has been observed and tracked in successive images and apparent 2D motions have been estimated. An *affine structure from motion* algorithm, [37], recovers 3D shape and motion of the rigid object with respect to the camera in two steps.

The first step in this algorithm is factorization: factoring a low-rank measurement matrix of apparent 2D motions into two skinny matrices of shape and motion. The original rank-3 algorithm is explained in [105, 37], and much faster rank-1 algorithms can be found in [2]. The factorization obtained is unique up to an arbitrary 3D affine transformation. It is very simple and is based on an SVD decomposition.

On the second step, the *Euclidean metric upgrade*, [37], is performed when taking into account the rigidity constraints. These additional constraints are derived by assuming that the affine camera model is restricted to be either orthographic, weak perspective, paraperspective, or fixed-intrinsic affine. These constraints are non-linear.

As properly noted in [37], the second step need not be performed for purely affine methods. These methods do not require camera calibration since the corresponding transformation of the image coordinates can be folded into the overall affine deformation of the object. This can be very advantageous for vision systems for which calibration parameters vary rapidly in time.

This fact motivates us to define a similarity measure for 3D shapes that will be invariant up to an arbitrary 3D affine transformation.

2.3 Why consider 2D affine transformations for 2D shapes?

When working with 2D shapes obtained directly from images, the 2D affine distortion model is commonly used. Under this model the 2D shapes of the same object are assumed to be related by an affine transformation. This approximation is very widely used in the literature in many image processing applications, [58].

For example, when imaging sensors are modeled as a weak perspective projection of the 3D world onto the 2D image plane and when the observed objects are planar, it is known that captured 2D images will be related by a 2D affine transformation, [85].

This fact motivates us to define a similarity measure for 2D shapes that will be invariant up to an arbitrary 2D affine transformation.

2.4 The affine space $\mathcal{A}(\mathcal{V})$ and its symmetry group $\mathcal{GA}(\mathcal{V})$

Now we start by presenting background material on the affine space in a group theoretic framework, [25, 38], as it will be required further in this dissertation. The presentation in this section follows [38].

As we mentioned in Section 2.1, an affine space is a vector space that has forgotten its origin. In order to forget its origin, we need to distinguish between points (particles) and vectors (forces acting on particles). We proceed as follows.

Definition: A group $\langle \mathcal{G}, + \rangle$ is said to act on a set X from the right when there is a map $\dot{+}_\wr : X \times \mathcal{G} \to X$ such that $\forall \mathbf{x} \in X$ and $\forall g, h \in \mathcal{G}$: $\mathbf{x} \dot{+}_\wr 0 = \mathbf{x}$ and $(\mathbf{x} \dot{+}_\wr g) \dot{+}_\wr h = \mathbf{x} \dot{+}_\wr (g + h)$. In this case, $\mathcal{G}$ is called a *"transformation group."* The map $\dot{+}_\wr$ is called a *"right group action."* Moreover, if for any two elements $\forall \mathbf{x}, \mathbf{y} \in X$, there is a unique $g \in \mathcal{G}$ such that $\mathbf{x} \dot{+}_\wr g = \mathbf{y}$, we say that group action $\dot{+}_\wr$ is *"transitive and faithful"* and we write $g \triangleq (\mathbf{y} \dot{-}_\wr \mathbf{x}) \in \mathcal{G}$.

Definition: Let group $\mathcal{G}$ act on set X by $\dot{+}_\wr : X \times \mathcal{G} \to X$. Then we say that two elements $\forall \mathbf{x}_1, \mathbf{x}_2 \in X$ are $\mathcal{G}$-equivalent and we write $\mathbf{x}_1 \sim_{\mathcal{G}} \mathbf{x}_2$ if $\exists g \in \mathcal{G} : \mathbf{x}_1 \dot{+}_\wr g = \mathbf{x}_2$. The *"left coset"* of $\mathcal{G}$ with respect to some $\mathbf{x} \in X$ is a set of all $\mathcal{G}$-equivalent elements to $\mathbf{x}$ and

is given by $\mathbf{x}\mathcal{G} \triangleq \{\hat{\mathbf{x}} : \hat{\mathbf{x}} \in X, \hat{\mathbf{x}} \sim_{\mathcal{G}} \mathbf{x}\} = \{\mathbf{x} \dot{+}_{\curvearrowright} g : g \in \mathcal{G}\}$. Sometimes $\mathbf{x}\mathcal{G}$ is also called the *"orbit"* of a point $\mathbf{x}$ and is denoted by $\lceil \mathbf{x} \rfloor_G$. It is clear that orbits form a partition of X, i.e., every element of X belongs to one and only one orbit. Orbits are sometimes also called *"$\mathcal{G}$-equivalency classes."* The set of all orbits of X under the action of group $\mathcal{G}$ is denoted by $X/\mathcal{G}$ and is called the *"quotient set"* of the action.

Definition: An *"affine space"* is a triplet $\langle P, \mathcal{V}, \dot{+}_{\curvearrowright} \rangle$ of an abstract set of points (particles) P, of a vector space of tangent vectors (forces) $\mathcal{V}$, and a transitive faithful group action $\dot{+}_{\curvearrowright} : P \times \mathcal{V} \to P$.

As we will see later in Section 3.5, the group action $\dot{+}_{\curvearrowright}$ in the definition of the affine space is called the *"exponential map."* Moreover, since group action $\dot{+}_{\curvearrowright}$ is transitive and faithful, it uniquely defines a difference of two points as a vector: $\forall \mathbf{p}, \mathbf{q} \in P \; v \triangleq (\mathbf{q} \dot{-}_{\curvearrowright} \mathbf{p}) \in \mathcal{V}$ and $\dot{-}_{\curvearrowright}$ is called the *"logarithmic map."* Intuitively, v is a vector from point $\mathbf{p}$ to point $\mathbf{q}$, and is often called a *"free vector."* We say that we equipped the set of points P with the vector space structure of $\mathcal{V}$. One particular case is when $P = \mathcal{V}$ and $\dot{+}_{\curvearrowright}$ is a canonical addition of vectors. We say that $\mathcal{A}(\mathcal{V}) = \langle \mathcal{V}, \mathcal{V}, + \rangle$ is a canonical affine space of a vector space $\mathcal{V}$. In particular, let $\mathbb{F}^p$ be a finite dimensional vector space over a field $\mathbb{F}$. We denote affine space over a field $\mathbb{F}$ by $\mathcal{A}^p(\mathbb{F})$. When the field is the set $\mathbb{R}$ of real numbers then we simply write $\mathcal{A}^p$.

Let $\mathcal{A} = \langle P, \mathcal{V}, \dot{+}_{\curvearrowright} \rangle$ be an affine space, and let us fix a point $\mathbf{p} \in P$. Consider a vector space $\mathcal{T}_{\mathbf{p}}\mathcal{A} = \{(\mathbf{q} \dot{-}_{\curvearrowright} \mathbf{p}) \in \mathcal{V} : \mathbf{q} \in P\} = (P \dot{-}_{\curvearrowright} \mathbf{p})$ of free vectors with origin at $\mathbf{p}$. This space is called the *"tangent space"* to $\mathcal{A}$ at point $\mathbf{p}$. Clearly, a set of points P now has a vector space structure of $\mathcal{T}_{\mathbf{p}}\mathcal{A}$. In particular, one can now consider linear combinations of vectors in $\mathcal{T}_{\mathbf{p}}\mathcal{A}$. It turns out that the linear combinations of vectors with coefficients summing to 1 are independent of a choice of $\mathbf{p}$. They are called barycentric combinations of vectors originating from point $\mathbf{p}$. Forgetting the origin $\mathbf{p}$ one now can define a *"barycentric combination of points,"* which is invariant of the origin.

Definition: Let $(\mathbf{p}_i)_{i \in I}$ be a family of points in P, and let $(\lambda_i)_{i \in I}$ be a family of scalars

summing to 1. Then we define a *"barycenter"* to be a point $\mathbf{m} \in P$ such that:

$$\mathbf{m} = \mathbf{p} \dot{+} \sum_{i \in I} \lambda_i \cdot (\mathbf{p_i} \dot{-} \mathbf{p}) \tag{2.1}$$

This definition is invariant of a choice of $\mathbf{p} \in P$, so we simply write: $\mathbf{m} = \sum_{i \in I} \lambda_i \mathbf{p_i}$.

As one can see, this definition shows how to compute the barycentric combination of points without fixing an origin. This is exactly how we forget the origin in the affine space. Now, we need to introduce a group of symmetries in the affine space. The *"symmetry group"* of a geometric object, as we mentioned earlier, is, intuitively, the set of all transformations that can be applied to it without changing the object. These transformations are non-degenerate maps that preserve the properties of the underlying space. In particular, for the affine space, such maps are bijective maps preserving the barycentric combination of points.

Definition: Let $\mathcal{A} = \langle P, \mathcal{V}, \dot{+} \rangle$ be an affine space. We say that function $f : P \to P$ is an *"affine map"* iff f preserves barycenters. In other words, for any family of points $\forall (\mathbf{p}_i)_{i \in I}$ and a family of scalars $\forall (\lambda_i)_{i \in I}$ such that $\sum_{i \in I} \lambda_i = 1$ we must have $f(\sum_{i \in I} \lambda_i \mathbf{p_i}) = \sum_{i \in I} \lambda_i f(\mathbf{p_i})$

Remark: A set of bijective affine maps on $\mathcal{A}(\mathcal{V})$ is clearly a group under map composition. This group is called the *"affine group"* and is denoted by $\mathcal{GA}(\mathcal{V})$. This group is a symmetry group of the affine space. It has a known structure — a semidirect product [17] of the general linear group $\mathcal{GL}(\mathcal{V})$ and a group of translations $\mathcal{T}(\mathcal{V})$.

We now define a semidirect product and give some of its properties that will be useful later. These definitions follow [17].

Definition: A *"normal subgroup"* $\mathcal{N}$ of a group $\mathcal{G}$ is a subgroup invariant under conjugation. In other words, $\forall g \in \mathcal{G}$, the set $g\mathcal{N}g^{-1} \subseteq \mathcal{N}$. We write $\mathcal{N} \lhd \mathcal{G}$.

The subgroup $\mathcal{N}$ is normal if and only if $g\mathcal{N} = \mathcal{N}g$ for $\forall g \in \mathcal{G}$. Then it is clear that all subgroups of an abelian group (i.e., a group for which the elements commute) are normal.

Definition: Let $\mathcal{N} \lhd \mathcal{G}$. Then the set $\mathcal{H} = \mathcal{G}/\mathcal{N} \triangleq \{g\mathcal{N} : g \in \mathcal{G}\}$ is a subgroup of $\mathcal{G}$; it is called the *"quotient group"* or the *"factor group."* Intuitively, $\mathcal{H}$ is a group that collapses the normal subgroup $\mathcal{N}$ to the identity element. Elements of the quotient group are $\mathcal{N}$-equivalency classes.

Definition: Let us consider two arbitrary groups $\mathcal{N}$ and $\mathcal{H}$ and let $\mathcal{A}ut(\mathcal{N})$ be a group of automorphisms from $\mathcal{N}$ onto itself. Given a group homomorphism $\alpha : \mathcal{H} \to \mathcal{A}ut(\mathcal{N})$, the cartesian product $\mathcal{G} = \mathcal{N} \times \mathcal{H}$ equipped with the binary operation $\circledast$ and given by $(n_1, h_1) \circledast (n_2, h_2) = (n_1 \cdot \alpha_{(h_1)}(n_2), h_1 \cdot h_2)$ is a group called the *"semidirect product"* of $\mathcal{N}$ and $\mathcal{H}$ with respect to α. We write $\mathcal{G} = \mathcal{N} \rtimes_\alpha \mathcal{H}$.

Remark: The direct product of groups is a special form of the semidirect product. To see this consider an example of $\alpha_{(h)}(n) = n$. Then we have $(n_1, h_1) \circledast (n_2, h_2) = (n_1 \cdot n_2, h_1 \cdot h_2)$, which is a direct product of $\mathcal{N}$ and $\mathcal{H}$. Another example would be to take $\alpha_{(h)}(n) = hnh^{-1}$. Let us use $\cong$ to denote the isomorphism of groups and let $\mathcal{G} = \mathcal{N} \rtimes_\alpha \mathcal{H}$. Then the set $\mathcal{N} \times \{1_\mathcal{H}\} \cong \mathcal{N}$ forms a normal subgroup of $\mathcal{G}$, while the set $\{1_\mathcal{N}\} \times \mathcal{H} \cong \mathcal{H}$ forms a subgroup of $\mathcal{G}$ where $1_\mathcal{N}$ and $1_\mathcal{H}$ are identity elements of $\mathcal{N}$ and $\mathcal{H}$ respectfully. Since now $\mathcal{N} \lhd \mathcal{G}$ we can consider the factor group $\mathcal{G}/\mathcal{N} \cong \mathcal{H}$. We emphasize to the reader here that when the group $\mathcal{G}$ is formed as a semidirect product of its normal subgroup $\mathcal{N}$ and its subgroup $\mathcal{H}$, we should factor out the normal subgroup first, i.e., in general $\mathcal{G}/\mathcal{H}$ will not be a group, while $\mathcal{G}/\mathcal{N}$ will be a group. We will see later why this is important for the affine group.

Proposition 2.4.1. *Let $\mathcal{A}(\mathcal{V})$ be an affine space. Then its affine group $\mathcal{GA}(\mathcal{V})$ can be decomposed into the semidirect product of two groups as follows:*

$$\mathcal{GA}(\mathcal{V}) = \mathcal{T}(\mathcal{V}) \rtimes_\alpha \mathcal{GL}(\mathcal{V}) \tag{2.2}$$

where the abelian group $\mathcal{T}(\mathcal{V})$ of translations with elements $\delta \in \mathcal{T}(\mathcal{V})$ is acting on $\mathbf{x} \in \mathcal{V}$ from the right by $\delta : \mathcal{V} \to \mathcal{V}$, $\mathbf{x} \mapsto \mathbf{x} + \delta$. The group $\mathcal{GL}(\mathcal{V})$ of general linear transformations of $\mathcal{V}$ with elements $A \in \mathcal{GL}(\mathcal{V})$ is acting on $\mathbf{x} \in \mathcal{V}$ from the left by $A : \mathcal{V} \to \mathcal{V}$, $\mathbf{x} \mapsto A\mathbf{x}$. The group

homomorphism $\alpha : \mathcal{GL}(\mathcal{V}) \to \mathcal{A}ut(\mathcal{T}(\mathcal{V}))$ *is acting in the obvious way* $\alpha_{(A)} : \mathcal{T}(\mathcal{V}) \to \mathcal{T}(\mathcal{V})$, $\delta \mapsto \alpha_{(A)}(\delta) \triangleq A{\cdot}\delta$.

One can easily see that $\mathcal{T}(\mathcal{V}) = \mathcal{V}$ and $\mathcal{GL}(\mathcal{V}) = \mathcal{A}ut(\mathcal{V})$, so we have a semidirect product of the special form $\mathcal{GA}(\mathcal{V}) = \mathcal{V} \rtimes \mathcal{A}ut(\mathcal{V})$. Also we are going to show how the affine group $\mathcal{GA}(\mathcal{V})$ is acting on $\mathcal{V}$ in a more familiar to the reader form. Let $\delta \in \mathcal{T}(\mathcal{V})$ and $A \in \mathcal{GL}(\mathcal{V})$ as before. Then the tuple $(\delta, A) \in \mathcal{GA}(\mathcal{V})$ is acting on $\mathbf{x} \in \mathcal{V}$ by $(\delta, A) : \mathcal{V} \to \mathcal{V}$ as follows:

$$\mathbf{x} \longmapsto \mathbf{x}(\delta, A) \triangleq A{\cdot}\mathbf{x} + \delta \tag{2.3}$$

This is the most common representation for affine transformations. When $\mathcal{A}(\mathcal{V}) = \mathcal{A}^p$ then A is a p-by-p non-singular matrix and δ is a p-by-1 vector with elements in $\mathbb{R}$. In this particular case we denote $\mathcal{GA}(\mathcal{V}) = \mathcal{GA}_{\mathrm{p}}$ and we write the decomposition as $\mathcal{GA}_{\mathrm{p}} = \mathcal{T}_p \rtimes \mathcal{GL}_{\mathrm{p}}$.

So far we showed how affine transformations are acting on just one point in the affine space. Now let us consider a family of points undergoing the same affine transformation. We discuss it next.

2.5 From one particle to a set of particles

Let us recall from Section 2.1 that we consider an object to be a set of particles, just like in physics. We model particles to reside in the affine space $\mathcal{A}^p$. Then the geometric object is an unsorted set of points in the affine space $\mathcal{A}^p$. Also, a symmetry group of the object that will keep it unchanged is $\mathcal{GA}_{\mathrm{p}}$. First we give several definitions.

Definition: A *"shape"* of an object is an unsorted family of points in the affine space $\mathcal{A}^p$, possibly with repetitions in order to emphasize more important ones:

$$\mathbf{s} = \Big(\mathbf{p}_i \in \mathcal{A}^p\Big)_{i \in I} \tag{2.4}$$

As a shorthand notation, we will write $\mathbf{s} \Subset \mathcal{A}^p$ where $\Subset$ denotes the fact that $\mathbf{s}$ is a family of points, possibly with repetitions, and not just a subset of $\mathcal{A}^p$. Let shape $\mathbf{s}$ consist of a finite family of $n = |\mathbf{s}|$ points. Then we refer to $\mathbf{s}$ to be an *"n-shape"*.

As a side, we repeat that throughout the dissertation we do not require a family of points to be distinct, i.e., allowing repetitions of points is up to the particular implementation. For example, one could attach integer masses to each particle in the object, and then repeatedly include the point into the family of points according to its mass. Instead of mass one could use a quantized intensity value of the particle. This is not the only way to incorporate the variable density of an object. For example, one could sample points more densely at places where there is more mass.

Now, without loss of generality let us assume that all the shapes we consider contain the same number of points n. We talk in Chapter 6 on comparing shapes with different number of points, or even doing multi-resolution comparison for different values of n. We define the set of all shapes of size n as follows.

Definition: A *"set of n-shapes"* is a set of all families of points from $\mathcal{A}^p$ of size n:

$$\mathcal{SS}_{n,p} \triangleq \{\mathbf{s} : \mathbf{s} \Subset \mathcal{A}^p, |\mathbf{s}| = n\} \tag{2.5}$$

We would like to equip the set of n-shapes $\mathcal{SS}_{n,p}$ with a geometry in order to be able to compare different shapes to each other. However, this cannot be done directly on the set $\mathcal{SS}_{n,p}$, since each shape is a family of abstract points. We proceed as follows. Recall that $\mathcal{A}^p = \langle P, \mathcal{V}, + \rangle$ where $P = \mathcal{V} = \mathbb{R}^p$. Let us fix an arbitrary point $\mathbf{o} \in P$. Then, as discussed in Section 2.4, our affine space is given by the vector space structure of its tangent space $\mathcal{T}_{\mathbf{o}}\mathcal{A}^p$. Further, we can fix some basis $\mathbf{e} = \left\{e^1, \cdots, e^p\right\}$ for the tangent space $\mathcal{T}_{\mathbf{o}}\mathcal{A}^p$. Informally, we fix a coordinate system and we attach a transformed copy of $\mathbb{R}^p$ at $\mathbf{o}$. The tuple $(\mathbf{o};\mathbf{e})$ is called a *"frame"* on the affine space $\mathcal{A}^p$. Now, every point $\mathbf{p} \in \mathcal{P}_{\mathrm{n}}$ is represented by a free vector $v = (\mathbf{p} - \mathbf{o}) \in \mathcal{T}_{\mathbf{o}}\mathcal{A}^p$, while the free vector v itself is represented by a $p \times 1$ vector $\mathbf{x} = \mathbf{e}v \in \mathbb{R}^p$ of coordinates with respect to the basis $\mathbf{e}$. In particular, a shape $\mathbf{s} \in \mathcal{SS}_{n,p}$ will be given representation with respect to the frame $(\mathbf{o};\mathbf{e})$ by a family of vectors in $\mathbb{R}^p$:

$$\mathbf{v} \triangleq \mathbf{v}_{\mathbf{s}}(\mathbf{o};\mathbf{e}) = \left(\mathbf{x}_i \triangleq \mathbf{e}\,(\mathbf{p}_i - \mathbf{o}) \in \mathbb{R}^p\right)_{i \in I} \Subset \mathbb{R}^p \tag{2.6}$$

Fig. 2.1: Different views $\mathbf{v_1}$, $\mathbf{v_2}$, $\mathbf{v_3}$ for the same shape $\mathbf{s}$ of the kettle

It is clear that as we vary a reference frame $(\mathbf{o};\mathbf{e})$, we will obtain different representations $\mathbf{v_s}(\mathbf{o};\mathbf{e})$ of the same shape $\mathbf{s} \in \mathcal{SS}_{n,p}$. We refer to these representations as *"views"* of the n-shape $\mathbf{s}$, or simply *"n-views."*

In Figure 2.1, as an example, we illustrate three different views for the same shape of a kettle. Each view is a representation of the kettle obtained with respect to a different reference frame. As one can see, each view is composed of a set of particles of the same mass that are shown as black dots. Each particle is a vector in $\mathbb{R}^p$ (with $p = 2$ in our particular case) and will have (x, y)-coordinates in the image plane. The view is just an unsorted family of size n of these (x, y) coordinates.

Lets consider all possible n-shapes $\mathbf{s} \in \mathcal{SS}_{n,p}$ and obtain all possible views for each shape. Then we can define a set of n-views as follows.

Definition: The *"set of n-views"* is the set of all families of vectors from $\mathbb{R}^p$ of size n:

$$\mathcal{SV}_{n,p} \triangleq \{\mathbf{v} : \mathbf{v} \Subset \mathbb{R}^p, |\mathbf{v}| = n\} \tag{2.7}$$

It is clear that shape views lie in this set, i.e. $\mathbf{v_s}(\mathbf{o};\mathbf{e}) \in \mathcal{SV}_{n,p}$. Recall that group $\mathcal{GA}_{\mathrm{p}}$ is acting on $\mathbb{R}^p$ as given in Equation (2.3). Then we can define a group action of $\mathcal{GA}_{\mathrm{p}}$ on $\mathcal{SV}_{n,p}$ to be a point-wise group action. We give the definition below.

Definition: Group $\mathcal{GA}_{\mathrm{p}}$ is acting on the right on $\mathcal{SV}_{n,p}$ by group action $* : \mathcal{SV}_{n,p} \times \mathcal{GA}_{\mathrm{p}} \to \mathcal{SV}_{n,p}$ such that $\forall \mathbf{v} \in \mathcal{SV}_{n,p}$ and $(\delta, A) \in \mathcal{GA}_{\mathrm{p}}$ we have $*(\delta, A) : \mathcal{SV}_{n,p} \to \mathcal{SV}_{n,p}$ as follows:

$$\mathbf{v} \longmapsto \mathbf{v} * (\delta, A) = \Big(A{\cdot}\mathbf{x} + \delta\Big)_{\mathbf{x}\in\mathbf{v}} \tag{2.8}$$

This group action definition has a natural explanation. Our object is rigid, so all points in the object will be undergoing the same affine transformation.

Since we have a properly defined group action, now it makes sense to talk about a quotient set $\mathcal{SV}_{n,p}/\mathcal{GA}_{\mathrm{p}}$. It is not hard to see that this quotient set is nothing else but $\mathcal{SS}_{n,p}$. We state the result below.

Lemma 2.5.1. *A set of n-shapes is a quotient set of group action $\mathcal{GA}_p$ on a set of n-views:*

$$\mathcal{SS}_{n,p} = \mathcal{SV}_{n,p}/\mathcal{GA}_p \tag{2.9}$$

As a result, n-shapes are orbits of equivalent n-views $\mathbf{s} = \lceil \mathbf{v} \rfloor_{\mathcal{GA}_{\mathrm{p}}}$ under the action of $\mathcal{GA}_{\mathrm{p}}$.

Note that we cannot work directly with a set of n-views $\mathcal{SV}_{n,p}$ and define a geometry on it. Next, we are going to show how we represent these views in a matrix form.

2.6 From sets to matrices and relation to permutation distortions

Consider a shape view $\mathbf{v}$ given by Equation (2.6). By definition, it is an unordered set of n distinct $p \times 1$ vectors. How do we represent this set numerically? Without loss of generality, let us assume the indexing set to be $I = \{1, \ldots, n\}$. Then any indexing order will be given by permutations of elements of I.

Definition: A *"permutation"* is a bijection from a finite set onto itself. In particular if the set under consideration is $\{1, \ldots, n\}$, then we can write $\pi : \{1, \ldots, n\} \to \{1, \ldots, n\}$. Associated with a permutation is the $n \times n$ *"permutation matrix:"*

$$P_\pi \triangleq \begin{bmatrix} e_{\pi(1)}^T \\ \vdots \\ e_{\pi(n)}^T \end{bmatrix} \tag{2.10}$$

with e_i being the i-th column vector in the $n \times n$ identity matrix. A set of all permutation matrices $\mathcal{P}_{\mathrm{n}} \triangleq \{P_\pi : \pi \text{ is a permutation } \}$ forms a finite group called the *"permutation group."*

Let $\mathbf{v}$ be given by Equation (2.6) and let π be some permutation. Then we define a stacking operator Stack : $\mathcal{SV}_{n,p} \times \mathcal{P}_{\mathrm{n}} \to \mathbb{R}^n_p$ from a set of n-views to $n \times p$ tall-skinny matrices:

$$\mathbf{X}_\pi \triangleq \mathrm{Stack}(\mathbf{v}, \pi) = \begin{bmatrix} \mathbf{x}^T_{\pi(1)} \\ \dots \\ \mathbf{x}^T_{\pi(n)} \end{bmatrix} = P_\pi \begin{bmatrix} \mathbf{x}^T_1 \\ \dots \\ \mathbf{x}^T_n \end{bmatrix} \tag{2.11}$$

Definition: Define the matrix $\mathbf{X}_\pi$ in Equation (2.11) to be an *"n-configuration"* of the underlying n-shape $\mathbf{s}$ in Equation (2.4) and the image space of the Stack operator to be the *"set of n-configurations"* $\mathcal{SC}_{n,p} \triangleq \mathrm{Stack}(\mathcal{SV}_{n,p}, \mathcal{P}_{\mathrm{n}})$.

Clearly, $\mathcal{SC}_{n,p}$ consists of all possible $n \times p$ matrices, so it will coincide with $\mathbb{R}^n_p$.

Once again, as evident from Equation (2.11), the group $\mathcal{P}_{\mathrm{n}}$ of permutations acts on the set $\mathcal{SC}_{n,p}$, and the quotient set is $\mathcal{SV}_{n,p}$. The result is stated below.

Lemma 2.6.1. *A set of n-views is the quotient set of the action of $\mathcal{P}_n$ on the set of n-configurations:*

$$\mathcal{SV}_{n,p} = \mathcal{SC}_{n,p}/\mathcal{P}_n \tag{2.12}$$

Next, we motivate why permutation distortions arise from a more natural perspective.

2.7 Why do we care about permutation distortions?

The shapes that we consider are families of feature points. We do not restrict how the feature points are selected. They can either be points scanned on a dense grid, or a set of points of interest with strong directional gradients in time, or points on object boundaries, or correspond to any other possible choice.

Independently of how feature points are selected, when the shape is distorted, for example, by an affine shape distortion, the feature points that comprise the shape move to different locations on the image plane. Since the image is scanned in a fixed order, such as the raster scan order, by the input device, the feature points of affine distorted shapes are not guaranteed to be read in the same order, thus adding another degree of ambiguity.

We identify this type of shape irregularity as a permutation distortion. To illustrate this point by example, look at Figure 2.1. The very first point scanned in the first view will be located on the handle of the kettle. The same point will not be scanned first in the second and third views.

When feature points are labeled, then they can be ordered according to their labels in order to avoid permutation distortions. The permutation of feature points is commonly addressed as the well-known feature *correspondence* problem, see [73, 72].

2.8 Affine-Permutation group and its decomposition

We start by recaping what we have so far. Let us look at the first three blocks of the diagram depicted in Figure 2.2. We started from an abstract set of n-shapes $\mathcal{SS}_{n,p}$ where each shape is a family of n abstract points in $\mathcal{A}^p$, not necessarily unique. This is shown on the first two lines of the first block in Figure 2.2. Then, in order to go from an abstract notion of a shape to a numeric representation, we showed that by fixing a frame in $\mathcal{A}^p$ we obtain a view of the shape. Also we showed that a set of shapes $\mathcal{SS}_{n,p}$ is a quotient of a set of views $\mathcal{SV}_{n,p}$ by the group of symmetries $\mathcal{G}A_{\mathrm{p}}$. The group of symmetries $\mathcal{G}A_{\mathrm{p}}$ itself is given as a semidirect product $\mathcal{G}A_{\mathrm{p}} = \mathcal{T}_p \rtimes \mathcal{G}L_{\mathrm{p}}$. Each view is a family of vectors from $\mathbb{R}^p$. This is shown in the second block and the arrow connecting the first and the second blocks of Figure 2.2.

Views are families of size n of vectors in $\mathbb{R}^p$. We defined a stacking operation that takes these vectors and stacks them together into an $n \times p$ configuration matrix. Since the ordering of the vectors is undefined, each view is uniquely determined by its configuration matrix up to a permutation. So a set of views $\mathcal{SV}_{n,p}$ is a quotient of a set of configurations $\mathcal{SC}_{n,p}$ by the group of permutations $\mathcal{P}_{\mathrm{n}}$. This is shown in the third block and the arrow connecting the second and the third blocks of Figure 2.2.

Now we are going to show that, in fact, we have a combined affine-permutation group, which is a direct product of $\mathcal{G}A_{\mathrm{p}}$ and $\mathcal{P}_{\mathrm{n}}$ groups. This group acts on a set of configurations $\mathcal{SC}_{n,p}$, and its quotient is a set of shapes $\mathcal{SS}_{n,p}$.

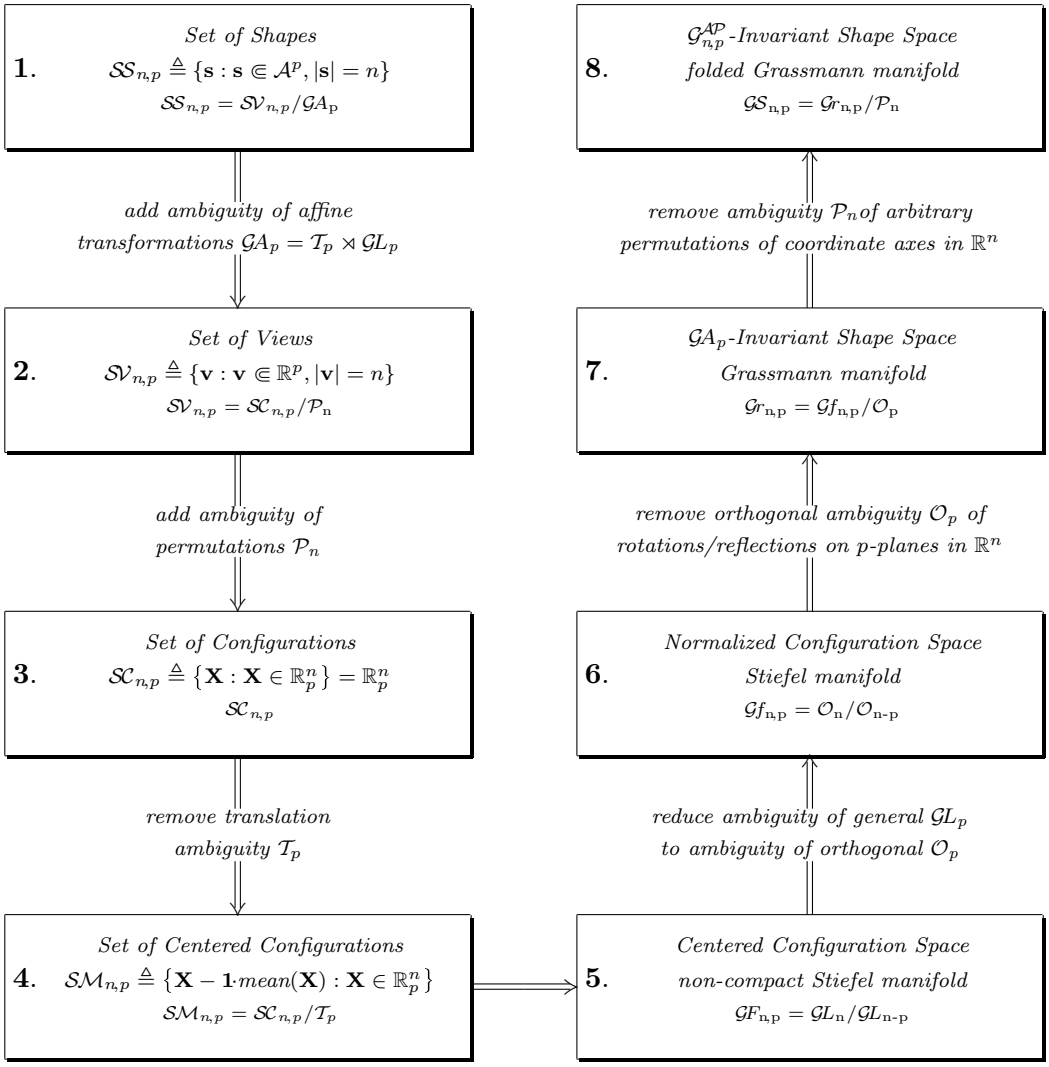

Fig. 2.2: From a set of shapes to shape space

Definition: We define an *"affine-permutation group"* to be a direct product:

$$\mathcal{G}^{\mathcal{AP}}_{\mathrm{n,p}} \triangleq \mathcal{G}A_{\mathrm{p}} \times \mathcal{P}_{\mathrm{n}} = (\mathcal{T}_p \rtimes \mathcal{G}L_{\mathrm{p}}) \times \mathcal{P}_{\mathrm{n}} \tag{2.13}$$

Let $(\delta, A, P) \in \mathcal{G}^{\mathcal{AP}}_{\mathrm{n,p}}$ be a triplet where $(\delta, A) \in \mathcal{G}A_{\mathrm{p}}$ and $P \in \mathcal{P}_{\mathrm{n}}$. Now we are going to define how this triplet is acting on a set of configurations.

Definition: Group $\mathcal{G}^{\mathcal{AP}}_{\mathrm{n,p}}$ is acting on the right on $\mathcal{SC}_{n,p}$ by group action $* : \mathcal{SC}_{n,p} \times \mathcal{G}^{\mathcal{AP}}_{\mathrm{n,p}} \rightarrow \mathcal{SC}_{n,p}$ such that $\forall \mathbf{X} \in \mathcal{SC}_{n,p}$ and $(\delta, A, P) \in \mathcal{G}^{\mathcal{AP}}_{\mathrm{n,p}}$ we have $*\,(\delta, A, P) : \mathcal{SC}_{n,p} \rightarrow \mathcal{SC}_{n,p}$ as follows:

$$\mathbf{X} \longrightarrow \mathbf{X} * (\delta, A, P) = P\left(\mathbf{X}A^T + \mathbf{1}\delta^T\right) = P\mathbf{X}A^T + \mathbf{1}\delta^T \tag{2.14}$$

where $\mathbf{1}$ is is an $n \times 1$ vector of ones. In fact, one can easily verify that this group action is defined correctly. We start by examining Equation (2.3). If we apply a transpose operation, then it becomes $\mathbf{x}^T \rightarrow \mathbf{x}^T A^T + \delta^T$. Now recall that the stacking operator, given in Equation (2.11), stacks transposed vectors $\mathbf{x}^T$ on top of each other into the matrix $\mathbf{X}$. Then, when the same affine transformation is applied to all the points stacked into the matrix $\mathbf{X}$, it will be of the form $\mathbf{X} \rightarrow P\mathbf{X}A^T + \mathbf{1}\delta^T$ where P is the permutation from the stacking operation.

Lemma 2.8.1. *The set of shapes is the quotient set of the action of $\mathcal{G}^{\mathcal{AP}}_{n,p}$ on a set of configurations:*

$$\mathcal{SS}_{n,p} = \mathcal{SC}_{n,p}/\mathcal{G}^{\mathcal{AP}}_{n,p} \tag{2.15}$$

As shown in the lemma, we can go from a set of configurations $\mathcal{SC}_{n,p}$ to a set of shapes $\mathcal{SS}_{n,p}$ by acting on $\mathcal{SC}_{n,p}$ with the group $\mathcal{G}^{\mathcal{AP}}_{\mathrm{n,p}}$. However, it is easier to break down the action of the group $\mathcal{G}^{\mathcal{AP}}_{\mathrm{n,p}}$ into a set of consecutive actions of the more primitive groups $\mathcal{T}_p$, $\mathcal{G}L_{\mathrm{p}}$ and $\mathcal{P}_{\mathrm{n}}$, as given in Equation (2.13). Moreover, the easiest way of equipping a set of shapes with the geometry is by using a quotient geometry obtained as a result of factoring out these groups from a Riemannian manifold with a geometry already defined on it. For this reason, the group of translations $\mathcal{T}_p$ needs to be factored out before the group $\mathcal{G}L_{\mathrm{p}}$ since it is a normal subgroup of $\mathcal{G}A_{\mathrm{p}}$. As to the group of permutations $\mathcal{P}_{\mathrm{n}}$, it can be factored out first, second or third since it commutes with $\mathcal{G}A_{\mathrm{p}}$. We discuss our choice next.

2.9 Shape space reduction as a quotient of manifolds

Figure 2.2, starting from the third block, shows the reduction process of the set of configurations $\mathcal{SC}_{n,p}$ into the shape space $\mathcal{GS}_{\mathrm{n,p}}$. Each factor space is obtained by acting on the space above it with some group action and forming orbits. Recall that orbits are the equivalency classes with respect to the group action. For each orbit we could identify and keep the canonical representatives, the way it is done in [45]. Alternatively, we can stay within the space of orbits, as they allow to work with them by picking an arbitrary member of the orbit without first finding a canonical member. This is a preferred method in cases when canonical members cannot be easily defined or computed due to ambiguities and noise in the input data. Stated differently, more informally, group actions are the distortions we are trying to remove. A process of finding a canonical member is essentially an estimation of these distortions. Whenever the estimation can not be reliably performed, instead of estimating the distortions we simply factor them out.

We choose to first factor out the group of translations $\mathcal{T}_p$, as shown by transition from block three to block four. On this step, we replace $\lceil \mathbf{X} \rfloor_{\mathcal{T}_p}$ with the canonical member defined by $\mathbf{X_c} : \mathbf{X_c} \in \lceil \mathbf{X} \rfloor_{\mathcal{T}_p}$, $\mathrm{mean}(\mathbf{X_c}) = 0$. We recall that a shape is represented by an $n \times p$ configuration matrix $\mathbf{X} \in \mathcal{SC}_{n,p}$; $\mathrm{mean}(\mathbf{X})$ is defined to be an $1 \times p$ row vector containing the mean value of each column of $\mathbf{X}$. We factor out translations via the centering operation $\mathbf{X_c} \triangleq \mathbf{X} - \mathbf{1} \cdot \mathrm{mean}(\mathbf{X})$ where $\mathbf{1}$ is an $n \times 1$ vector of ones as before. We refer to the quotient set $\mathcal{SM}_{n,p} \triangleq \mathcal{SC}_{n,p}/\mathcal{T}_p$ to be a set of centered configurations.

$\mathcal{SM}_{n,p}$ is not a Riemannian manifold. We place additional restriction on the elements of $\mathcal{SM}_{n,p}$ in order to make it a manifold. We will require that the $n \times p$ centered configuration matrices $\mathbf{X_c}$ be of rank p. Note that the centering operation could reduce the rank, so the restriction of having rank p is properly placed on centered configurations. Let us analyze now what shapes are excluded by this restriction. It is clear that for $p = 2$ the only shapes we will not be able to look at will be the ones for which points lie on the same line, and for $p = 3$ we will not be able to look at planar shapes. In general, this restriction tells us that we cannot study shapes of smaller dimensions by looking at them in higher dimensions,

which is a very plausible assumption, since these excluded shapes can always be studied by embedding them in a space of proper dimension. In Figure 2.2, by placing this restriction we are moving from the fourth block to the fifth block. We explain next how centered configurations of rank p form a Riemannian manifold.

A centered configuration matrix is of size $n \times p$ and has rank p. We can view this matrix as a linearly independent p-frame in $\mathbb{R}^n$. To complete the matrix to full basis of $\mathbb{R}^n$ we can choose any non-singular $(n-p)$-frame. Thus, we can realize it as a quotient of general linear groups, which are also Lie groups, [110, 16], as $\mathcal{G}F_{\text{n,p}} = \mathcal{G}L_{\text{n}}/\mathcal{G}L_{\text{n-p}}$. We note that $\mathcal{G}F_{\text{n,p}}$ obtained this way is a smooth manifold. It is frequently referred to as the *"non-compact Stiefel manifold."* For background information we refer the reader to [16]. It is shown in the fifth block of Figure 2.2. We should note that centered configurations are just a subset of $\mathcal{G}F_{\text{n,p}}$. In fact, it turns out that they form a subspace of $\mathcal{G}F_{\text{n,p}}$, but for now we can skip this discussion, as it carries no importance. We can assume that we are working with the complete $\mathcal{G}F_{\text{n,p}}$ space.

Next, we construct $\mathbf{Y} \cong \lceil \mathbf{X_c} \rfloor_{\mathcal{N}} \in \mathcal{G}f_{\text{n,p}}$ from $\mathbf{X_c} \in \mathcal{G}F_{\text{n,p}}$ by replacing the diagonal matrix of the singular values in its compact SVD decomposition with the identity. Replacing the singular values with ones makes $\mathbf{Y}$ to be orthonormal ($\mathbf{Y}^T\mathbf{Y} = \mathbf{I}$). The obtained space $\mathcal{G}f_{\text{n,p}} = \mathcal{O}_{\text{n}}/\mathcal{O}_{\text{n-p}}$ is a space of orthonormal p-frames, and is called the *"compact Stiefel manifold"* or simply *"Stiefel manifold,"* [16].

We note that in the literature both $\mathcal{G}F_{\text{n,p}}$ and $\mathcal{G}f_{\text{n,p}}$ are frequently called the Stiefel manifold. We will refer to $\mathcal{G}F_{\text{n,p}}$ as the non-compact Stiefel manifold and to $\mathcal{G}f_{\text{n,p}}$ as the Stiefel manifold. Even though we could work directly with $\mathcal{G}F_{\text{n,p}}$, we prefer to project first $\mathcal{G}F_{\text{n,p}}$ onto $\mathcal{G}f_{\text{n,p}}$ and then work with orthonormal matrices $\mathbf{Y} \in \mathcal{G}f_{\text{n,p}}$. The difference between the two is that in $\mathcal{G}F_{\text{n,p}}$ we work with any linearly independent p-frames, while in $\mathcal{G}f_{\text{n,p}}$ we work with orthonormal p-frames. This is the sixth block of Figure 2.2.

We should note here that the use of an SVD decomposition is closely related with the Gram-Schmidt orthogonalization process. It allows us to reduce the non-compact Lie group $\mathcal{G}L_{\text{n}}$ to the compact Lie group $\mathcal{O}_{\text{n}}$. Moreover, it can be shown that $\mathcal{O}_{\text{n}} = \mathcal{G}L_{\text{n}}/\mathcal{P}DUT_{\text{n}}$ where

$\mathcal{PDUT}_{\mathrm{n}}$ is the matrix subgroup consisting of all upper triangular matrices with positive diagonal entries, see [8].

In order to eliminate the orientation ambiguity, we take a Lie group $\mathcal{O}_{\mathrm{p}}$ and consider a quotient space $\mathcal{G}f_{\mathrm{n,p}}/\mathcal{O}_{\mathrm{p}}$. Each element in the quotient space is the set of all possible rotated and reflected versions of $\mathbf{Y}$ and is given by $\lceil\mathbf{Y}\rfloor_{\mathcal{O}_{\mathrm{p}}} = \{\mathbf{Y}V : V \in \mathcal{O}_{\mathrm{p}}\}$. The manifold $\mathcal{G}r_{\mathrm{n,p}} = \mathcal{G}f_{\mathrm{n,p}}/\mathcal{O}_{\mathrm{p}}$ obtained in this way is called the *Grassmann manifold*, [16]. It can also be defined as the space of all p-dim subspaces in $\mathbb{R}^n$. Under this definition, if $\mathbf{p}$ is a point on the Grassmann manifold $\mathcal{G}r_{\mathrm{n,p}}$ then $\mathbf{p} = \mathrm{span}(\mathbf{Y})$. Both definitions are equivalent, but we will work with the orbits $\lceil\mathbf{Y}\rfloor_{\mathcal{O}_{\mathrm{p}}}$. We will use a shorthand symbol $\lceil\mathbf{Y}\rfloor$, omitting the group $\mathcal{O}_{\mathrm{p}}$ as its subindex when it is clear that the action is by the group $\mathcal{O}_{\mathrm{p}}$.

We emphasize that the seventh block on the right-hand side of Figure 2.2 is the Grassmann manifold $\mathcal{G}r_{\mathrm{n,p}} = \mathcal{G}f_{\mathrm{n,p}}/\mathcal{O}_{\mathrm{p}}$. This manifold is of great importance, as it provides us with an affine-invariant shape space. We study its geometry in Chapter 4.

Note how we took the group of affine transformations $\mathcal{G}A_{\mathrm{p}}$ and decomposed it as a semidirect product $\mathcal{G}A_{\mathrm{p}} = \mathcal{T}_p \rtimes \mathcal{G}L_{\mathrm{p}}$. We first factored out the group of translations $\mathcal{T}_p$, and then separately from it we factored out the group of general linear transformations $\mathcal{G}L_{\mathrm{p}}$. We ended up in the Grassmann manifold $\mathcal{G}r_{\mathrm{n,p}}$. Alternatively, we could have considered $\mathcal{G}A_{\mathrm{p}}$ as a sub-group of $\mathcal{G}L_{\mathrm{p+1}}$ using the standard trick of representing affine transformations by $(p+1)\times(p+1)$ matrices of the form:

$$\begin{bmatrix} A^T & 0 \\ \delta^T & 1 \end{bmatrix} \in \mathcal{G}L_{\mathrm{p+1}} \tag{2.16}$$

where $(\delta, A) \in \mathcal{G}A_{\mathrm{p}}$. Then we could use the standard trick of going from Euclidean space $\mathbb{R}^p$ of shape view coordinates to its projective counterpart by adding an additional coordinate being equal to 1. This results in augmenting $n \times p$ configuration matrices with an extra column of ones and going to augmented $n \times p+1$ configuration matrices. These augmented configuration matrices can be represented as points of the non-compact Stiefel manifold of one dimension higher $\mathcal{G}F_{\mathrm{n,p+1}}$. Then we can go to the Grassmann manifold of one dimension higher $\mathcal{G}r_{\mathrm{n,p+1}}$ in the usual way. The paper, given in [11], takes this approach combined

with our framework, see [98], in order to achieve an invariance to affine transformations. The authors in [11] avoid the centering operation by factoring out the affine group all at once. This approach generates a number of problems, making it less attractive. First of all, an affine-invariant shape space will no longer be the complete Grassmann manifold. It will be just its subset. There will be lots of points on the Grassmann manifold $\mathcal{G}r_{\mathrm{n,p+1}}$ that will not correspond to any shapes. Then, by performing various geometric operations on $\mathcal{G}r_{\mathrm{n,p+1}}$, such as following geodesics or computing sample mean, we might end up at a point of $\mathcal{G}r_{\mathrm{n,p+1}}$ that is not a valid shape. There are also other disadvantages. An additive noise affecting the position of the feature points in a shape view will not transfer nicely to the Grassmann manifold because of an extra coordinate. To summarize, we strongly discourage from using the approach.

Now we go back to Figure 2.2 and summarize the final step. It allows arbitrary permutations of n feature points, which are row permutations of the configuration matrices. This ambiguity is eliminated by taking a finite group of permutations $\mathcal{P}_{\mathrm{n}}$ and folding the Grassmann manifold into $\mathcal{G}r_{\mathrm{n,p}}/\mathcal{P}_{\mathrm{n}}$. This group does not act freely, [16]. Thus, $\mathcal{G}r_{\mathrm{n,p}}/\mathcal{P}_{\mathrm{n}}$ is no longer a manifold, and we cannot realize a geometry in $\mathcal{GS}_{\mathrm{n,p}}$ as a quotient geometry of $\mathcal{G}r_{\mathrm{n,p}}/\mathcal{P}_{\mathrm{n}}$. However, it is still possible to define a geometry in the neighborhoods of points and perform computations of similarity when points are sufficiently close to each other. So we will abuse notation and still call $\mathcal{G}r_{\mathrm{n,p}}/\mathcal{P}_{\mathrm{n}}$ a folded space. This folded space is shown in the block eight of Figure 2.2 and is discussed in further details in Chapter 5.

2.10 Shapes as points of a Riemannian manifold

The definition of a shape is very abstract, and there is no easy injective map between the set of all shapes $\mathcal{SS}_{n,p}$ and an Euclidean space. In previous sections we introduced an injection that maps shapes into points of a Riemannian manifold.

Since shapes are represented by points on a manifold, we can no longer use algorithms for the Euclidean space and have to create equivalent algorithms for the Riemannian space. There are a few important building blocks for most of the algorithms of interest to us:

- ***efficient numerical representation of shapes***
- ***computation of distance between 2 shapes***
- ***barycentric combination of k shapes***
- ***sample mean and higher order statistics***
- ***moving from a shape in some tangent direction***

In Chapters 4, 5 we derive formulas for all the above mentioned algorithm building blocks. We also show how these basic concepts can be applied to classification by applying them to multi-author handwritten digit recognition in Chapter 6.

2.11 Summary

In this chapter we addressed issues related to shape representation. We started by motivating the reader why affine distortions of a rigid object is an import class of distortions to consider. We considered a model under which an object is a set of particles, while particles live in the affine space. We gave group theoretic definitions of the affine space and its symmetry group — the affine group. Then we gave an abstract definition of a shape as an unsorted family of particles in the affine space. We also defined shape views to be a visualization of a shape under a given affine transformation, and shape configuration matrices to be numeric representations of these views under a given pair of an affine transformation and an ordering (permutation) of the particles. Then we derived under the group theoretic framework the combined affine-permutation group $\mathcal{G}_{\mathrm{n,p}}^{\mathcal{AP}}$ and gave its decomposition into three primitive groups $\mathcal{T}_p$, $\mathcal{GL}_{\mathrm{p}}$ and $\mathcal{P}_{\mathrm{n}}$. We gave definitions of group actions and quotient spaces formed by group actions. We saw that we can sequentially factor out each one of the primitive groups in order to achieve injective shape representation by points of a Riemannian manifold.

We noted that we form an equivalency classes whenever we consider factor spaces. In order to factor out an action of the group under the consideration we can then either obtain a canonical member for each of equivalency classes or stay within the space of orbits and derive formulas that are invariant to the choice of the representative of each equivalency

class. Whenever computation of a canonical member is simple and is resilient to noise, we chose to find a canonical member. However, whenever there is no easy noise insensitive algorithm for canonical member calculation, we stay within the space of orbits.

We started by factoring out the group of translation $\mathcal{T}_p$. On this step we find a canonical member by means of centering the configuration matrix around the center of mass of an object. This operation is resilient to the noise added to the location of particles.

Then we reduced an action of the non-compact general linear group $\mathcal{G}L_{\mathrm{p}}$ to an action of the compact orthogonal group $\mathcal{O}_{\mathrm{p}}$. This is a projection operation from the non-compact Stiefel manifold $\mathcal{G}F_{\mathrm{n,p}}$ to the compact Stiefel manifold $\mathcal{G}f_{\mathrm{n,p}}$, and is resilient to noise.

Next we factored out the orthogonal group $\mathcal{O}_{\mathrm{p}}$ that lead us to the affine-invariant shape space representation by points on the real Grassmann manifold. On this step we chose to stay within the space of orbits because an operation of finding a canonical member is not an easy one, and is noise sensitive, [45]. A typical approach that is taken by in the literature for this step is defining principal axes of the shape and then un-rotating the shape to align its principal axes with their canonical position. As one can see, even small noise can introduce a large change in a position of principal axes causing discontinuities in the similarity measure. We avoid this approach by staying within the space of orbits and developing formulas on the Grassmann manifold that are invariant to the choice of the representative of each equivalency class.

Finally, by factoring out the permutation group $\mathcal{P}_{\mathrm{n}}$, we obtained the affine-permutation-invariant shape space representation by points on the folded Grassmann manifold. On this step we chose to find a canonical member because action of the finite group $\mathcal{P}_{\mathrm{n}}$ is not free, as it will be explained in Section 5.2.

In addition, in this chapter we summarized the important algorithm building blocks one needs to address in order to be able to perform various algorithms in the Riemannian spaces.

3. EMBEDDED RIEMANNIAN MANIFOLDS

3.1 Preliminaries

In this chapter we consider embedded Riemannian manifolds. The key points that will be addressed in this chapter are:

- ***Vector fields, integral curves, and flows — Section 3.2***
- ***Parallel transport, geodesic vector fields and geodesic flows — Section 3.3***
- ***Riemannian metric, Levi-Civita connection, and distance — Section 3.4***
- ***Exponential and logarithmic maps — Section 3.5***
- ***Expected values and barycentric combinations — Section 3.6***
- ***Orthonormal group $\mathcal{O}_n$ as a Riemannian manifold — Section 3.7***

We start by explaining the notions of tangent spaces and the tangent bundle of the manifold, vector fields and flows, geodesics and associated geodesic vector fields, the notions of distance. Also we give a more intuitive representation for the exponential map as an addition of a vector to a point, and its inverse — the logarithmic map — as a subtraction of two points. In addition we give various definitions for the mean on a Riemannian manifold and define a notion of barycentric combinations. Finally, we apply these notions on the example of the orthogonal group $\mathcal{O}_{\mathrm{n}}$ when it is considered as an embedded Riemannian manifold. These preliminaries and background is needed in the subsequent chapters of the thesis.

Now, before we continue further, we present key notions that we need from the theory of manifolds. The presentation of the material here is informal. It does not follow any of the sources, and in many instances we cannot even give direct references to the background material. As for general references, we refer the reader for the necessary background on

differentiable manifolds to classical textbooks on Riemannian geometry and Lie theory [68, 16, 23, 70, 103, 21, 13, 38, 8, 41, 49, 7, 29].

Riemannian manifolds consist of three layers. The first layer is a topological layer. Let $\mathcal{M}$ be a Riemannian manifold. Then first of all $\mathcal{M}$ possesses a topological structure. As part of its topological structure, we require $\mathcal{M}$ to be Hausdorff and second countable. Also $\mathcal{M}$ has to look locally, as a topological space, like Euclidean space. Manifolds inherit many of the local properties of Euclidean space. They are locally path-connected and locally compact. However, for example, global compactness is not guaranteed.

The second layer is the differential layer. It makes $\mathcal{M}$ a differentiable manifold. At this layer we require $\mathcal{M}$ to be a smooth manifold, which implies that derivatives of all orders exist. We talk about this layer in more detail in Sections 3.2-3.3.

Finally, the third layer is the Riemannian layer. This layer allows $\mathcal{M}$ to become a metric space, thus providing the ability for computing distances in between points. This layer is discussed in Section 3.4.

We also discuss the concepts of the exponential map and its inverse — the logarithmic map. We relate these concepts with the concepts of expected values and barycentric combinations. These are discussed in Sections 3.5-3.6.

Finally, in Section 3.7, we apply these notions on the example of the orthogonal group $\mathcal{O}_{\mathrm{n}}$ when it is considered as an embedded Riemannian manifold.

3.2 Vector fields, integral curves, and flows

In this section we define vector fields, integral curves, and flows. A very good introduction to these concepts can be found in [70].

Let $\mathcal{M}$ be a smooth differentiable manifold. We proceed with the following definitions.

Definition: *"Smooth curves"* on $\mathcal{M}$ are smooth mappings $\gamma : \mathcal{I} \subseteq \mathbb{R} \to \mathcal{M}$ from an open interval $\mathcal{I}$ in the time domain $\mathbb{R}$ to $\mathcal{M}$. They carry a way of natural differentiation in time $\gamma' = \frac{d\gamma}{dt}$. The derivative $\dot{\mathbf{Q}}_\tau = \gamma'(\tau)$ is called the *"tangent vector"* at point $\mathbf{Q}_\tau = \gamma(\tau)$.

Definition: *"Tangent space"* $\mathcal{T}_{\mathbf{Q}}\mathcal{M}$ to a point $\mathbf{Q} \in \mathcal{M}$ is defined to be a set of all tangent vectors at point $\mathbf{Q}$, i.e. a set of all derivatives $\boldsymbol{\Delta} = \gamma'(t)\big|_{t=\tau}$ of smooth curves $\gamma(t)$ with initial condition $\gamma(t)\big|_{t=\tau} = \mathbf{Q}$.

Definition: *"Tangent bundle"* of a manifold $\mathcal{M}$, denoted by $\mathcal{TM}$, is the disjoint union of the tangent spaces $\mathcal{T}_{\mathbf{Q}}\mathcal{M}$ to each point $\mathbf{Q} \in \mathcal{M}$. An element of $\mathcal{TM}$ is an ordered pair $(\mathbf{Q}, \boldsymbol{\Delta})$ where $\mathbf{Q} \in \mathcal{M}$ and $\boldsymbol{\Delta} \in \mathcal{T}_{\mathbf{Q}}\mathcal{M}$. The *"natural projection"* $\pi : \mathcal{TM} \to \mathcal{M}$ is defined to send a pair $(\mathbf{Q}, \boldsymbol{\Delta})$ to the base point $\mathbf{Q}$.

Definition: Let $\mathcal{U} \subseteq \mathcal{M}$ be an open subset of $\mathcal{M}$. A (smooth) *"vector field"* is a mapping that assigns smoothly to each point on $\mathcal{U}$ one of its tangent vectors. Specifically, a vector field on $\mathcal{U}$ is a smooth map:

$$\Delta : \mathcal{U} \longrightarrow \mathcal{TM} \;\; ; \quad \mathbf{Q} \longmapsto \Delta(\mathbf{Q}) = \Big(\mathbf{Q}, \boldsymbol{\Delta}_{\mathbf{Q}}\Big) \tag{3.1}$$

where $\boldsymbol{\Delta}_{\mathbf{Q}} \in \mathcal{T}_{\mathbf{Q}}\mathcal{M}$. In other words, $\forall \mathbf{Q} \in \mathcal{U}$ we have $\pi\big(\Delta(\mathbf{Q})\big) = \mathbf{Q}$. For this reason, when a vector field Δ is defined globally on $\mathcal{M}$, it is also called a *"section"* of the tangent bundle. Smooth vector fields Δ over the manifold $\mathcal{M}$ form a vector space denoted by $\mathcal{X}(\mathcal{M})$.

One should think of a vector field on $\mathcal{M}$ as the analog of vector fields in Euclidean space. Each vector field is simply a vector attached to each point of $\mathcal{M}$ such that it is tangent to $\mathcal{M}$ and varies continuously from point to point.

Definition: Suppose $\gamma : \mathcal{I} \subseteq \mathbb{R} \to \mathcal{M}$ is some curve and $\Delta \in \mathcal{X}(\mathcal{M})$ is a vector field. We define a *"restriction of a vector field"* Δ along curve γ to be given by:

$$\Delta\big|_{\gamma} : \mathbb{R} \longrightarrow \mathcal{TM} \;\; ; \quad t \longmapsto \Delta\big|_{\gamma}(t) = \Big(\gamma(t), \boldsymbol{\Delta}_{\gamma(\mathbf{t})}\Big) \tag{3.2}$$

One particular case of interest is when the restriction of the vector field along curve γ coincides with the derivative γ'. It is called the *"derivative vector field"* or simply the *"velocity vector field"* of the curve γ, and we write:

$$\dot{\gamma}\big|_{\gamma} : \mathbb{R} \longrightarrow \mathcal{TM} \;\; ; \quad t \longmapsto \dot{\gamma}\big|_{\gamma}(t) = \Big(\gamma(t), \gamma'(t)\Big) \tag{3.3}$$

So far we considered smooth curves γ, which generate velocity vector fields $\dot{\gamma}\big|_{\gamma}$ along the curve. Now, we would like to work backward by taking a vector field and seeking for a smooth curve at each point whose velocity vector field will locally coincide with the given vector field. This motivates us to proceed with the definition of integral curves and flows, and to study their relationship to vector fields.

Definition: Let Δ be a vector field on $\mathcal{M}$. An *"integral curve"* of Δ is a smooth curve $\gamma : \mathcal{I} \subseteq \mathbb{R} \to \mathcal{M}$ defined on an open interval $\mathcal{I}$ such that:

$$\dot{\gamma}\big|_{\gamma}(t) = \Delta\big|_{\gamma}(t) \qquad \forall t \in \mathcal{I} \tag{3.4}$$

Theorem 3.2.1 (Existence and uniqueness of maximal integral curves).
Let $\Delta \in \mathcal{X}(\mathcal{M})$ be a vector field on $\mathcal{M}$. For all points on the manifold $\forall \mathbf{Q} \in \mathcal{M}$, there exists a unique maximal open interval $\mathcal{I}_{\mathbf{Q}} \subseteq \mathbb{R}$ containing a neighborhood of 0 and a unique integral curve of Δ $\gamma_{\mathbf{Q}} : \mathcal{I}_{\mathbf{Q}} \to \mathcal{M}$ defined on this interval such that $\gamma_{\mathbf{Q}}(0) = \mathbf{Q}$. In other words, at each point there exists only one "maximal integral curve" *of Δ passing through it.*

> *Proof.* We give a sketch of the proof, which is based on the theory of ordinary differential equations. Observe that an integral curve is defined by the first order differential equation $\big(\gamma_{\mathbf{Q}}(t), \gamma'_{\mathbf{Q}}(t)\big) = \Delta\big|_{\gamma_{\mathbf{Q}}}(t)$, and the initial condition is given by $\gamma_{\mathbf{Q}}(0) = \mathbf{Q}$. Existence and uniqueness then follow from the Picard-Lindelf theorem for the solutions of ODEs with prescribed initial conditions, [86]. □

Now we define a flow generated by a vector field Δ. Informally speaking, it is a family of integral curves associated with a vector field. More precise definition is given below.

Definition: Consider a family of maximal integral curves $\{\gamma_{\mathbf{Q}} : \mathbf{Q} \in \mathcal{M}\}$ of Δ from the theorem above and define a map Θ_{Δ}:

$$\Theta_{\Delta} : \mathcal{D} \subseteq \mathbb{R} \times \mathcal{M} \to \mathcal{M} \;\; ; \quad \Theta_{\Delta}\big(t, \mathbf{Q}\big) \triangleq \gamma_{\mathbf{Q}}(t) \tag{3.5}$$

with the domain $\mathcal{D}$ being a maximal open subset of $\mathbb{R} \times \mathcal{M}$ such that values of $\gamma_{\mathbf{Q}}(t)$ exist. Then Θ_{Δ} is called a *"flow"* generated by Δ, while Δ is called an *"infinitesimal generator"* of the flow Θ_{Δ}. When the domain $\mathcal{D} = \mathbb{R} \times \mathcal{M}$ the flow is called a *"global flow"* on $\mathcal{M}$.

Theorem 3.2.2. *A "global flow" $\Theta_\Delta : \mathbb{R} \times \mathcal{M} \to \mathcal{M}$ is a one-parameter group action.*

Proof. One easily verifies this indeed. By definition $\Theta_\Delta(0, \mathbf{Q}) = \mathbf{Q}$. Also, it can be proven that it satisfies the transitivity as well $\Theta_\Delta\big(t, \Theta_\Delta(s, \mathbf{Q})\big) = \Theta_\Delta(t + s, \mathbf{Q})$. □

Since global flows are one-parameter group actions, we introduce a shorthand notation for writing them. We define:

$$\mathbf{Q} \dotplus t\Delta \triangleq \Theta_\Delta(t, \mathbf{Q}) \tag{3.6}$$

Every global flow generates a vector field, while a vector field generates a flow that might not be a global flow. We are primarily interested in vector fields that do generate global flows. We give the theorem that establishes a sufficient condition.

Definition: We say that a vector field is *"complete"* if it generates a global flow, or equivalently if each of its integral curves is defined for $\forall t \in \mathbb{R}$.

Theorem 3.2.3. *If a manifold $\mathcal{M}$ is compact, then every vector field on $\mathcal{M}$ is complete.*

Proof. The proof can be found in [70]. □

So, if we consider compact smooth manifolds, then every vector field will generate a global flow, and every global flow will have a corresponding vector field.

3.3 Notions of parallel transport and geodesic vector fields

In Euclidean spaces we can take the directional derivative of one vector field with respect to the other vector field. We generalize this notion for manifolds by defining a connection. In a way, a connection is a generalization to taking second order derivatives.

Definition: A *"connection"* or a *"covariant derivative"* of a vector field with respect to another vector field is a mapping $\nabla : \mathcal{X}(\mathcal{M}) \times \mathcal{X}(\mathcal{M}) \to \mathcal{X}(\mathcal{M})$ such that $(\Delta_2, \Delta_1) \mapsto \nabla_{\Delta_2}\Delta_1$ and has properties of the derivative, i.e., it is algebraically linear, additive, and obeys the product rule. Also, a covariant derivative of a vector field Δ along some curve $\gamma(t)$ is defined to be a restriction of it along the curve and we write $\frac{D}{dt}\Delta\big|_\gamma \triangleq \nabla_{\dot\gamma}\Delta\big|_\gamma$. In particular, we define the second order derivative of a curve to be $\ddot\gamma\big|_\gamma \triangleq \frac{D}{dt}\dot\gamma\big|_\gamma = \nabla_{\dot\gamma}\dot\gamma\big|_\gamma$

Definition: A vector field $\Delta \in \mathcal{X}(\mathcal{M})$ is said to be a *"parallel vector field"* if its covariant derivative is zero with respect to any other vector field, i.e. $\forall \Delta_2 \in \mathcal{X}(\mathcal{M}) : \nabla_{\Delta_2} \Delta = 0$. In particular, a vector field Δ is said to be parallel along some curve $\gamma : \mathcal{I} \to \mathcal{M}$ if its covariant derivative is zero everywhere along the curve, i.e.:

$$\frac{D}{dt}\Delta\big|_{\gamma} = 0|_{\gamma} \quad : \quad 0|_{\gamma}(t) = \Big(\gamma(t), \mathbf{0} \in \mathcal{T}_{\gamma(t)}\mathcal{M}\Big) \;\; \forall t \in \mathcal{I} \tag{3.7}$$

Theorem 3.3.1 (Parallel vector field generator). *Let $\boldsymbol{\Delta}$ be a tangent vector at point $\mathbf{Q}$, and let $\mathcal{U} \subset \mathcal{M}$ be an open neighborhood of $\mathbf{Q}$. There exist one and only one parallel vector field on $\mathcal{U}$ that extends a given tangent vector $\boldsymbol{\Delta}$ at point $\mathbf{Q}$.*

Proof. The proof can be found in any standard textbook on differential geometry, [68]. However, note that it is not possible, given a tangent vector at a point, to extend it to a parallel vector field globally across the manifold. As an example, consider the unit-sphere and try to extend, in parallel way, any given non-zero tangent vector at the north pole. There exist a unique minimal geodesic segment connecting the north pole to any other point on a sphere, except for the south pole. These geodesics will uniquely extend, in parallel way, the tangent vector at the north pole. However, the tangent vector at the south pole will be undefined since there is more than one minimal geodesic segment connecting the two poles together. Thus, there are multiple ways of parallel transporting the tangent to the south pole. This will be discussed in more details in Section 3.5 when we talk about cut-locus. □

Let $\Pi : \mathcal{TM} \to \mathcal{X}(\mathcal{M})$ denote the generating function of the parallel vector field. Then the parallel vector field Δ that extends a given tangent vector $\boldsymbol{\Delta}$ at point $\mathbf{Q}$ will be given by $\Delta = \Pi\big(\mathbf{Q}, \boldsymbol{\Delta}\big)$. This generating function is generally well defined only in the neighborhood of $\mathbf{Q}$. Now we can define the parallel transport along a curve.

Definition: Consider a curve $\gamma : \mathcal{I} \to \mathcal{M}$ and a tangent vector $\boldsymbol{\Delta}_0$ at the point $\mathbf{Q}_0 = \gamma(0)$. Then the restriction of the parallel vector field along the curve γ generated by these initial conditions gives rise to a notion of parallel transport. The tangent vector $\boldsymbol{\Delta}_0$ at point $\gamma(0)$ is said to be *"parallel transported"* along the curve γ to the tangent vector $\boldsymbol{\Delta}_t$ at point $\gamma(t)$ where $\big(\gamma(t), \boldsymbol{\Delta}_t\big) = \Pi\big(\mathbf{Q}_0, \boldsymbol{\Delta}_0\big)\big|_{\gamma}(t)$.

Now, we generalize the definition of a straight line, called a geodesic, to be a curve γ of zero acceleration. The requirement of the second order derivative to be zero comes from

physics if we consider the fact that if an object moves along some curve at a constant pace, its acceleration will be zero only if this curve is a straight line.

Definition: A curve γ is called a *"geodesic"* if and only if its velocity vector field $\dot{\gamma}\big|_\gamma$ is a parallel vector field. In other words, γ is a geodesic if and only if $\ddot{\gamma}\big|_\gamma = 0|_\gamma$. In this case we refer to the velocity vector field $\dot{\gamma}\big|_\gamma$ as a *"geodesic vector field"*.

From the definition of the geodesic, one can see that the equation for the geodesic is a second order differential equation. Thus, we should expect existence and uniqueness for the solution of the geodesic differential equation in some small neighborhood. The following theorem states this result.

Theorem 3.3.2 (Existence and uniqueness of geodesics).
For all points on the tangent bundle $\big(\mathbf{Q}_0, \mathbf{\Delta}_0\big) \in \mathcal{TM}$ *there exists a unique maximal open interval* $\mathcal{I} \subseteq \mathbb{R}$ *containing a neighborhood of* 0 *and a unique geodesic* $\gamma : \mathcal{I} \to \mathcal{M}$ *defined on this interval such that* $\dot{\gamma}\big|_\gamma(0) = \big(\mathbf{Q}_0, \mathbf{\Delta}_0\big)$. *This geodesic is called a* "maximal integral geodesic."

> *Proof.* According to a Theorem 3.3.1, $\Pi\big(\mathbf{Q}_0, \mathbf{\Delta}_0\big)$ will be the unique parallel vector field extending $\big(\mathbf{Q}_0, \mathbf{\Delta}_0\big) \in \mathcal{TM}$. Then according to the Theorem 3.2.1 there exists a unique maximal integral curve $\gamma : \mathcal{I} \to \mathcal{M}$ of the vector field $\Pi\big(\mathbf{Q}_0, \mathbf{\Delta}_0\big)$ where $\mathcal{I}$ is an open interval containing a neighborhood of 0 and such that $\gamma(0) = \mathbf{Q}_0$. Moreover, since γ is an integral curve, by definition we have $\dot{\gamma}\big|_\gamma = \Pi\big(\mathbf{Q}_0, \mathbf{\Delta}_0\big)\big|_\gamma$. This shows that the velocity field of γ is a parallel vector field, and so γ is a geodesic. Moreover, $\dot{\gamma}\big|_\gamma(0) = \Pi\big(\mathbf{Q}_0, \mathbf{\Delta}_0\big)\big|_\gamma(0) = \big(\mathbf{Q}_0, \mathbf{\Delta}_0\big)$, so it satisfies the initial conditions. This completes the proof. □

Alternatively, one could prove this theorem using the theory of second order differential equations with two initial conditions. However, from our presentation of the proof we can immediately see that the geodesics give rise to geodesic vector fields, which in turn generate local geodesic flows. These flows become global geodesic flow when the geodesic vector fields are complete. In particular, based on Theorem 3.2.3 the sufficient condition for this is when the manifold is compact. Recall that our shorthand notation for writing the one-parameter group action of the flows is $\mathbf{Q}_t = \mathbf{Q}_0 + t\Pi\big(\mathbf{Q}_0, \mathbf{\Delta}_0\big)$. This notation can further be simplified for the geodesic flows, since the geodesic vector field $\Pi\big(\mathbf{Q}_0, \mathbf{\Delta}_0\big)$ is fully determined by the

initial conditions $(\mathbf{Q}_0, \boldsymbol{\Delta}_0)$. In this particular case we simply write:

$$\mathbf{Q}_t = \mathbf{Q}_0 \dot{+}_\gamma t\boldsymbol{\Delta}_0 \tag{3.8}$$

Informally speaking, we defined the addition of the scaled vector to the point to be another point. This notation essentially tells us that point $\mathbf{Q}_t$ is reached at time t by traveling along the unique geodesic defined by the initial conditions $\big(\mathbf{Q}_0, \boldsymbol{\Delta}_0\big)$. It is also closely related to the exponential maps, and will be discussed in further detail in Section 3.5.

We illustrate the results of Theorems 3.3.1 and 3.3.2 through the example given in Figure 3.1. We start with some point $\mathbf{Q_0}$ and some tangent at this point $\dot{\mathbf{Q}}_0$. According to Theorem 3.3.2, this pair defines a locally unique geodesic curve γ with initial conditions $\gamma(0) = \mathbf{Q_0}$ and $\gamma'(0) = \dot{\mathbf{Q}}_\mathbf{0}$. This curve is illustrated in the figure by an arch. Point $\mathbf{Q_0}$ on the manifold $\mathcal{M}$ is illustrated as a vector from some hypothetical origin, while $\dot{\mathbf{Q}}_0$ is simply a tangent to the arch. As we travel along the arch from time $t = 0$ to time $t = \tau$, $\mathbf{Q}_0$ moves to $\mathbf{Q}_\tau$ while $\dot{\mathbf{Q}}_0$ parallel transports to $\dot{\mathbf{Q}}_\tau$ according to the Theorem 3.3.1. Essentially, vector $\dot{\mathbf{Q}}_0$ is attached with its base to $\mathbf{Q}_0$, and as we move $\mathbf{Q}_0$ along the curve, $\dot{\mathbf{Q}}_0$ travels with it. For this reason, when we derive a solution to the geodesic equation $\gamma(t)$, we also need to derive a solution for parallel transport of its velocity vector $\gamma'(t)$. Recall that the geodesic vector field is the velocity vector field of the geodesic γ along itself and is given by $\dot{\gamma}\big|_\gamma(t) = \big(\gamma(t), \gamma'(t)\big)$. It captures how both $\gamma(t)$ and $\gamma'(t)$ change simultaneously. So, when deriving a solution to the geodesic equation, we are looking for the equation of the geodesic vector field $\dot{\gamma}\big|_\gamma$.

Let us look again at Figure 3.1 and pick some arbitrary tangent $\boldsymbol{\Delta}_0$ at point $\mathbf{Q}_0$ that is not tangent to the curve $\gamma(t)$. According to Theorem 3.3.1, there exists a unique parallel vector field $\Pi\big(\mathbf{Q}_0, \boldsymbol{\Delta}_0\big)|_\gamma$ that parallel transports $\boldsymbol{\Delta}_0$ to $\boldsymbol{\Delta}_\tau$ along the geodesic curve γ. Finding a solution to $\Pi\big(\mathbf{Q}_0, \boldsymbol{\Delta}_0\big)|_\gamma$ gives a general form of the parallel transport.

Now we move to defining distances and their relationship to the geodesic vector fields.

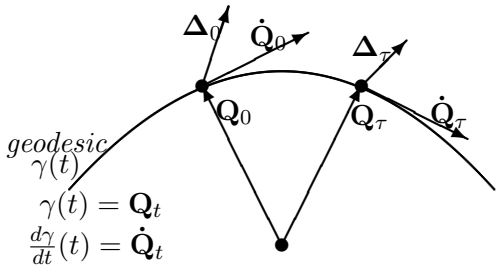

Fig. 3.1: Connecting points $\mathbf{Q}_0$ and $\mathbf{Q}_t$ by a geodesic $\gamma(t)$

3.4 Notion of intrinsic distance on a Riemannian manifold

Recall that the manifold $\mathcal{M}$ in general is not a vector space. Thus, we have no ability to equipping $\mathcal{M}$ with an inner product directly. However, tangent spaces at each point are vector spaces of the same dimension as the dimensionality of the manifold. Thus, equipping each tangent space with a norm and putting additional constrains allows us to turn $\mathcal{M}$ into a metric space.

Definition: A *"Riemannian metric"* on a smooth manifold $\mathcal{M}$ is a choice at each $\mathbf{Q} \in \mathcal{M}$ of a positive definite inner product $\langle \cdot, \cdot \rangle_{\mathbf{Q}}$ on $\mathcal{T}_{\mathbf{Q}}\mathcal{M}$ varying smoothly as $\mathbf{Q}$ traverses the manifold. A manifold $\mathcal{M}$ equipped with a Riemannian metric is called a *"Riemannian manifold."* As usual, the *"norm"* of some $\boldsymbol{\Delta} \in \mathcal{T}_{\mathbf{Q}}\mathcal{M}$ is defined to be:

$$\|\boldsymbol{\Delta}\|_{\mathbf{Q}} \triangleq \sqrt{\langle \boldsymbol{\Delta}, \boldsymbol{\Delta} \rangle_{\mathbf{Q}}} \tag{3.9}$$

Sometimes we will simply write $\|\boldsymbol{\Delta}\|$ implicitly implying that the norm is computed using the metric of the appropriate tangent space to which the tangent vector belongs.

Recall that a choice of the connection $\nabla : \mathcal{X}(\mathcal{M}) \times \mathcal{X}(\mathcal{M}) \to \mathcal{X}(\mathcal{M})$ is not unique on the differentiable manifold. However, once $\mathcal{M}$ is given a Riemannian manifold structure, there is only one symmetric connection. The result is stated below.

Definition: A connection ∇ is called *"compatible"* with the Riemannian metric if it preserves inner products during parallel transport.

Theorem 3.4.1 (Levi-Civita connection). *A Riemannian manifold possesses one and only one symmetric connection that is compatible with its metric. Such connection is called a* "Levi-Civita connection."

From now on, we always assume a manifold $\mathcal{M}$ to be a Riemannian manifold with the Levi-Civita connection ∇. We are now ready to discuss the property of geodesics to be locally the shortest path between points on the manifold.

Definition: Let $\gamma : [0, \tau] \to \mathcal{M}$ be a smooth curve on the Riemannian manifold. Then its length is given by:

$$L(\gamma) = \oint_0^\tau \Big\|\gamma'(t)\Big\|_{\gamma(t)} dt \tag{3.10}$$

Similarly, length of a piecewise smooth curve is a finite sum of the lengths of all pieces. A *"metric"* $\mathrm{d}\big(\mathbf{Q}_0, \mathbf{Q}_\tau\big)$ on $\mathcal{M}$ is constructed as an infimum of $L(\gamma)$ where γ is a piecewise smooth curve connecting the two points together.

Theorem 3.4.2 (Geodesic as a local distance minimizer). *Let $\gamma : \mathcal{I} \to \mathcal{M}$ be the maximal geodesic with initial conditions $\dot\gamma\big|_\gamma(0) = \big(\mathbf{Q}, \mathbf{\Delta}\big)$ as given in the Theorem 3.3.2. It is locally length minimizing, i.e., $\forall t \in \mathcal{I}$, there $\exists \epsilon > 0$ such that $\forall t_1, t_2 \in (t - \epsilon, t + \epsilon) \cap \mathcal{I}$ we have:*

$$d\Big(\gamma(t_1), \gamma(t_2)\Big) = \big\|\mathbf{\Delta}\big\|_{\mathbf{Q}} \cdot \big|t_1 - t_2\big| \tag{3.11}$$

Proof. The proof is very standard and is given in standard textbooks. Here we will show only a sketch since it is important for the reader to understand how distances are calculated. Consider the length of the geodesic curve between times $t_1 \le t_2$:

$$L\Big(\gamma(t_1), \gamma(t_2)\Big) = L(\gamma)\Big|_{t_1}^{t_2} = \oint_{t_1}^{t_2} \Big\|\gamma'(t)\Big\|_{\gamma(t)} dt \tag{3.12}$$

Since the curve $\gamma(t)$ is a geodesic, its velocity vector field $\gamma'(t)$ is the parallel vector field. At the same time, a Levi-Civita connection is compatible with the Riemannian metric and so preserves norms of parallel vector fields. Thus, the norm will be constant in the integration, and we can write:

$$L\Big(\gamma(t_1), \gamma(t_2)\Big) = \int_{t_1}^{t_2} \Big\|\gamma'(0)\Big\|_{\gamma(0)} dt = \Big\|\mathbf{\Delta}\Big\|_{\mathbf{Q}} \cdot (t_2 - t_1) \tag{3.13}$$

By analogy with physics, the quantity $\big\|\mathbf{\Delta}\big\|_{\mathbf{Q}}$ represents the velocity at which we are traveling. The acceleration is zero since we are traveling at constant speed along a straight path. Thus,

distance is simply velocity times time difference. At the same time, the path of zero acceleration will be the shortest path. So we can conclude that $\mathrm{d}\big(\gamma(t_1),\gamma(t_2)\big) = L\big(\gamma(t_1),\gamma(t_2)\big)$. □

Note that geodesics in general are not globally path minimizing, thus $\gamma(t_1)$ and $\gamma(t_2)$ should lie sufficiently close to each other.

3.5 Addition of a vector to a point and difference of two points

In this section we give definitions of the exponential map and its inverse — the logarithmic map. We relate these definitions to addition of a vector to a point and difference of two points and give various related definitions.

Recall that the geodesic flow with initial conditions $\big(\mathbf{Q},\boldsymbol{\Delta}\big) \in \mathcal{TM}$ is given by $\mathbf{Q}_t = \mathbf{Q} \dot{+} t\boldsymbol{\Delta}$. Suppose it exists at time $t = 1$. Then we traveled along the geodesic defined by $\big(\mathbf{Q},\boldsymbol{\Delta}\big)$ to point $\mathbf{Q}_1 = \mathbf{Q} \dot{+} \boldsymbol{\Delta}$ that is $\big\|\boldsymbol{\Delta}\big\|$ away from point $\mathbf{Q}$. This mapping has a special name — the exponential map.

Definition: Let $\mathbf{Q}$ be a point on the manifold $\mathcal{M}$. We define an *"exponential map"* to be:

$$\exp_{\mathbf{Q}} : \mathcal{V}_{\mathbf{Q}} \subseteq \mathcal{T}_{\mathbf{Q}}\mathcal{M} \longrightarrow \mathcal{M} \;\; ; \quad \boldsymbol{\Delta} \longmapsto \exp_{\mathbf{Q}}\big(\boldsymbol{\Delta}\big) \triangleq \mathbf{Q} \dot{+} \boldsymbol{\Delta} \tag{3.14}$$

with the domain $\mathcal{V}_{\mathbf{Q}}$ being a maximal open subset of $\mathcal{T}_{\mathbf{Q}}\mathcal{M}$ such that values $\mathbf{Q} \dot{+} \boldsymbol{\Delta}$ exist.

In other words, this map defines the addition of a point with a vector attached to this point to be another point that lies in the direction of the vector, the length of the vector away. In a way, the exponential map can be viewed as a way of achieving a linearization of the manifold around the point $\mathbf{Q}$.

Lemma 3.5.1. *The exponential map* $\exp_{\mathbf{Q}}$ *is a local diffeomorphism. It embedds a neighborhood* $\mathcal{TU}(\mathbf{0})$ *of the origin at* $\mathcal{T}_{\mathbf{Q}}\mathcal{M}$ *to a neighborhood* $\mathcal{U}(\mathbf{Q})$ *of* $\mathbf{Q}$ *in the manifold* $\mathcal{M}$.

Definition: Since $\exp_{\mathbf{Q}}$ is a local diffeomorphism, its inverse exists at least locally in a neighborhood of $\mathbf{Q}$, and is called the *"logarithmic map:"*

$$\log_{\mathbf{Q}} : \mathcal{U}_{\mathbf{Q}} \subseteq \mathcal{M} \longrightarrow \mathcal{T}_{\mathbf{Q}}\mathcal{M} \;\; ; \quad \mathbf{Q}_e \longmapsto \log_{\mathbf{Q}}\big(\mathbf{Q}_e\big) : \mathbf{Q}_e = \mathbf{Q} \dot{+} \log_{\mathbf{Q}}\big(\mathbf{Q}_e\big) \tag{3.15}$$

Since the exponential map is an addition operation, the logarithmic map can be viewed as a subtraction operation, which tells us that the difference of two points is a vector:

$$\log_{\mathbf{Q}}\left(\mathbf{Q}_e\right) = \mathbf{\Delta} \triangleq \mathbf{Q_e} \ \dot{-}_{\gamma} \mathbf{Q} \tag{3.16}$$

Now we study in more detail the existence and uniqueness of the logarithmic map.

Definition: We call a Riemannian manifold $\mathcal{M}$ *"geodesically complete,"* or simply *"complete,"* if, for all $\mathbf{Q} \in \mathcal{M}$, the exponential map $\exp_{\mathbf{Q}}$ is defined on all of $\mathcal{T}_{\mathbf{Q}}\mathcal{M}$.

Lemma 3.5.2. *A compact manifold is complete.*

Proof. Based on Theorem 3.2.3, every vector field of a compact manifold is complete, thus giving rise to a global geodesic flow. This guarantees that the exponential map is defined everywhere. This shows the manifold to be geodesically complete. □

Definition: A geodesic segment γ connecting two points $\mathbf{Q}_1, \mathbf{Q}_2 \in \mathcal{M}$ is said to be *"minimal"* if it is distance realizing, e.g. if its length $L(\gamma)$ is equal to the distance in between the points $\mathrm{d}(\mathbf{Q}_1, \mathbf{Q}_2)$.

Lemma 3.5.3 (existence of minimal geodesic segment). *If a (connected) Riemannian manifold $\mathcal{M}$ is complete, then every two points $\mathbf{Q}_1, \mathbf{Q}_2 \in \mathcal{M}$ can be joined by a minimal geodesic segment.*

Proof. See the theorem of Hopf-Rinow, [61]. □

So, if we consider compact smooth manifolds, then they are also complete, and thus exponential maps are defined everywhere. However, even though each $\exp_{\mathbf{Q}}$ is defined on the whole tangent space $\mathcal{T}_{\mathbf{Q}}\mathcal{M}$, it will in general not be a global diffeomorphism. In other words, the logarithmic map is multi-valued, and so the minimal geodesic segments do exist, but might not be unique. Now, we give additional definitions related to these concepts.

Definition: Let $\left(\mathbf{Q}, \mathbf{\Delta}\right) \in \mathcal{TM}$. The *"cut point"* of $\mathbf{Q}$ along the geodesic γ determined by $\left(\mathbf{Q},\mathbf{\Delta}\right)$ initial conditions is the first point on γ such that for any point beyond it γ is no longer the distance minimizer. The set $\mathcal{C}(\mathbf{Q})$ of these points along the geodesics emanating from $\mathbf{Q}$ is called the *"cut locus"* of $\mathbf{Q}$.

Definition: A neighborhood $\mathcal{U}(\mathbf{Q})$ of $\mathbf{Q} \in \mathcal{M}$ is said to be *"normal"* or *"star-convex"* if every $\mathbf{Q}_u \in \mathcal{U}(\mathbf{Q})$ can be joined to $\mathbf{Q}$ by a unique minimal geodesic segment fully contained in $\mathcal{U}(\mathbf{Q})$, [89].

Lemma 3.5.4. *Let* $\mathbf{Q}$ *be a point on the complete (connected) Riemannian manifold* $\mathcal{M}$. *Then the set* $\mathcal{I}(\mathbf{Q}) = \mathcal{M} \setminus \mathcal{C}(\mathbf{Q})$ *is an open normal neighborhood of* $\mathbf{Q}$. *Moreover,* $\mathcal{I}(\mathbf{Q})$ *is the maximal domain of the logarithmic map* $\log_{\mathbf{Q}}$.

Proof. See Sakai, [96]. □

The set $\mathcal{I}(\mathbf{Q})$ is called the *"interior"* at $\mathbf{Q}$. Its image via the logarithmic map

$$\mathcal{TI}(\mathbf{0}) \triangleq \log_{\mathbf{Q}} \big(\mathcal{I}(\mathbf{Q})\big) \tag{3.17}$$

is called the *"tangential interior."* In fact, $\mathcal{TI}(\mathbf{0})$ is the maximal open neighborhood of the origin at $\mathcal{T}_{\mathbf{Q}}\mathcal{M}$ that is diffeomorphic to the neighborhood $\mathcal{I}(\mathbf{Q})$ of $\mathbf{Q}$ in the manifold $\mathcal{M}$. Suppose the manifold is complete. Then for all the cut points the logarithmic map is multi-valued. We define

$$\mathcal{TC}(\mathbf{0}) \triangleq \log_{\mathbf{Q}} \big(\mathcal{C}(\mathbf{Q})\big) \tag{3.18}$$

to be the *"tangential cut locus."* It can also be viewed as a preimage of the cut locus in the tangent space $\mathcal{T}_{\mathbf{Q}}\mathcal{M}$ via the exponential map. For the points in the tangential cut locus the exponential map is not injective. Its image is the cut locus.

Informally speaking, the cut locus of $\mathbf{Q}$ is a border contour beyond which geodesic segments having $\mathbf{Q}$ as a startpoint are no longer distance minimizing. In fact, Helgason, see [52], has shown that the cut locus of a compact connected Riemannian manifold with the Lie group structure is stratified, i.e. it is the disjoint union of its smooth submanifolds. Because of this, cut locus is often conveniently visualized as a star-shaped contour. Another reason for visualizing it as a star-shaped contour is because of star-convexivity of the interior $\mathcal{I}(\mathbf{Q})$.

So far we saw that the set $\mathcal{I}(\mathbf{Q})$ is the largest open star-convex neighborhood of $\mathbf{Q}$. In additition to star-convex neighborhoods sometimes it is useful to consider convex neighborhoods. We define them next.

Definition: A neighborhood $\mathcal{U}(\mathbf{Q})$ of $\mathbf{Q} \in \mathcal{M}$ is said to be *"convex"* if every two $\mathbf{Q}_u, \mathbf{Q}_v \in \mathcal{U}(\mathbf{Q})$ can be joined by a unique minimal geodesic segment fully contained in $\mathcal{U}(\mathbf{Q})$.

One particular example of open convex neighborhoods is a geodesic ball $\mathcal{B}_r(\mathbf{Q})$ of sufficiently small radius r. First, we start by defining a geodesic ball. Then, we talk about upper bounds on the radius of a geodesic ball for it to be convex. A geodesic ball is defined as an image of an open ball $\mathcal{TB}_r(\mathbf{Q})$ in the tangent space $\mathcal{T}_{\mathbf{Q}}\mathcal{M}$ via the exponential map, where an open ball in the tangent space is the interior of a hyper-sphere of dimension $\dim(M) - 1$ and radius r. One needs to define the largest radius r such that $\mathcal{B}_r(\mathbf{Q})$ is convex. Clearly, r is the distance from $\mathbf{Q}$ to the cut locus $\mathcal{C}(\mathbf{Q})$. The formal definition is given below.

Definition: Let $\mathbf{Q} \in \mathcal{M}$. The radius of the largest open convex ball about the origin in $\mathcal{T}_{\mathbf{Q}}\mathcal{M}$ is called the *"radius of injectivity"* of $\mathcal{M}$ at $\mathbf{Q}$. The infimum of injectivity radii at each point $\mathbf{Q} \in \mathcal{M}$ is called the radius of injectivity of $\mathcal{M}$.

The following lemma relates the radius of injectivity to the cut locus.

Lemma 3.5.5. *For a complete Riemannian manifold $\mathcal{M}$, the injectivity radius at $\mathbf{Q} \in \mathcal{M}$ is the infimum of lengths of the geodesic segments connecting $\mathbf{Q}$ with each of its cut points.*

Now we go back to the discussion on the difference of two points on the manifold. Let $\mathcal{M}$ be a compact Riemannian manifold and let $\mathbf{Q}, \mathbf{Q}_e \in \mathcal{M}$ be two points. We saw that if point $\mathbf{Q}_e$ belongs to the interior of point $\mathbf{Q}$, i.e. $\mathbf{Q}_e \in \mathcal{I}(\mathbf{Q})$, then the logarithmic map is diffeomorphic. Thus, $\boldsymbol{\Delta} = \mathbf{Q}_e \ \text{-}\!\!\gamma\, \mathbf{Q}$ exists and is well defined. Moreover, the pair $\left(\mathbf{Q}, \boldsymbol{\Delta}\right)$ will uniquely define the path minimizing geodesic that connects the two points together. Thus, in fact, this equation tells us that given two points on the manifold, we would like to solve the two-point boundary value problem for the path minimizing geodesic connecting these two points. As for the distance, we will have:

$$d\left(\mathbf{Q}_e, \mathbf{Q}\right) = \left\|\boldsymbol{\Delta}\right\| = \left\|\mathbf{Q}_e \ \text{-}\!\!\gamma\, \mathbf{Q}\right\| \tag{3.19}$$

Finally, we should note that for some special matrix Lie groups, endowed with a canonical bi-invariant metric, the exponential map and the logarithmic map coincide with the

definition of the matrix exponential and the matrix logarithm, see [7, 8]. This is exactly where the names of these maps come from. The matrix exponential is well defined, however the matrix logarithm is not well defined and it is hard to compute. For example, see [39].

3.6 Expected values and barycentric combinations

In this section we first start by motivating the reader for the definition of barycentric combinations. Then we define them formally and discuss conditions for their existence and uniqueness. For further reading we refer the reader to books on statistical inference on Riemannian manifolds, see [5, 84, 65].

Let $\mathcal{M}$ be a complete connected Riemannian manifold. Suppose Ψ is a sample space and let $\mathcal{Q} : \Psi \to \mathcal{M}$ be a random object on $\mathcal{M}$. Take some point $\mathbf{Q} \in \mathcal{M}$ and consider its interior $\mathcal{I}(\mathbf{Q})$. It will be a star-convex open neighborhood of $\mathbf{Q}$, and so $\log_{\mathbf{Q}}$ is a diffeomorphism onto the tangent space $\mathcal{T}_{\mathbf{Q}}\mathcal{M}$. Then for each $x \in \Psi$ the value $\mathcal{Q}(x) \in \mathcal{M}$ can be mapped onto the tangent space at $\mathbf{Q}$ as:

$$\log_{\mathbf{Q}} \mathcal{Q}\big(x\big) \triangleq \log_{\mathbf{Q}} \Big(\mathcal{Q}(x)\Big) = \mathcal{Q}(x) \ \text{-}_{\curlywedge}\, \mathbf{Q} \tag{3.20}$$

In other words, the object $\mathcal{Q} \ \text{-}_{\curlywedge}\, \mathbf{Q} = \log_{\mathbf{Q}} \mathcal{Q} : \Psi \to \mathcal{T}_{\mathbf{Q}}\mathcal{M}$ is a random vector in the tangent space $\mathcal{T}_{\mathbf{Q}}\mathcal{M}$, and so one can consider an expected value $\mathrm{E}\big(\mathcal{Q} \ \text{-}_{\curlywedge}\, \mathbf{Q}\big)$. Then to find the expected value of $\mathcal{Q}$, calculated around the point $\mathbf{Q}$, one simply applies the exponential map to go back from the tangent space to the manifold. So we will have:

$$\mathrm{E}_{\mathbf{Q}}\big(\mathcal{Q}\big) = \mathbf{Q} \mathbin{\dot{+}_{\curlywedge}} \mathrm{E}\big(\mathcal{Q} \ \text{-}_{\curlywedge}\, \mathbf{Q}\big) \tag{3.21}$$

When the sample space Ψ is continuous, we can write:

$$\mathrm{E}_{\mathbf{Q}}\big(\mathcal{Q}\big) = \mathbf{Q} \mathbin{\dot{+}_{\curlywedge}} \int_{\Psi} \Big(\mathcal{Q}(x) \ \text{-}_{\curlywedge}\, \mathbf{Q}\Big)\mu(x)dx \ \ ; \quad \int_{\Psi} \mu(x)dx = 1 \tag{3.22}$$

Similarly, for the discrete case we have:

$$\mathrm{E}_{\mathbf{Q}}\big(\mathcal{Q}\big) = \mathbf{Q} \mathbin{\dot{+}_{\curlywedge}} \sum_{i \in \Psi} \mu_i \Big(\mathcal{Q}_i \ \text{-}_{\curlywedge}\, \mathbf{Q}\Big) \ \ ; \quad \sum_{i \in \Psi} \mu_i = 1 \tag{3.23}$$

One expects that assuming the points $\mathcal{Q}(x)$ in the constellation lie sufficiently close to each other and that both $\mathcal{Q}$ and μ are well behaved measurable maps on $\mathcal{M}$, then the expected value should exist and be unique. Moreover, it should not depend on the choice of the point $\mathbf{Q}$ around which we diffeomorphically mapped the manifold to the vector space. Recall Equation (2.1) from Section 2.4 for the affine space, and compare it to Equation (3.23). For the affine space, choice of the point around which we diffeomorphically mapped the manifold to the vector space was not important. The final value did not depend on it. Unfortunately, this is not, in general, the case for Riemannian manifolds. However, under certain conditions, the expected value exists and is unique. For this reason, similar to the affine case, one would like to simply write:

$$\mathrm{E}\big(\mathcal{Q}\big) \triangleq \int_{\Psi} \mathcal{Q}(x)\mu(x)dx \;\; ; \quad \int_{\Psi} \mu(x)dx = 1 \tag{3.24}$$

and for the discrete case:

$$\mathrm{E}\big(\mathcal{Q}\big) \triangleq \sum_{i\in\Psi} \mu_i \mathcal{Q}_i \;\; ; \quad \sum_{i\in\Psi} \mu_i = 1 \tag{3.25}$$

In fact, we do not even need to require the map μ to be a valid probability distribution, i.e. to be non-negative. It is sufficient to require that μ be bounded and sum to 1. In this case $\mathrm{E}\big(\mathcal{Q}\big)$ is commonly called the weighted *"barycentric combination"* of the constellation of points $\mathcal{Q}$ with weights μ. Now we present more formal definitions.

Definition: Consider the following scalar cost function:

$$\mathcal{C}_{\mathcal{Q}}\big(\mathbf{Q}\big) \triangleq \frac{1}{2}\int_{\Psi} \Big\| \mathcal{Q}(x) \;\overset{\mathbf{Q}}{\ominus}\; \mathbf{Q} \Big\|^2 \mu(x)dx \tag{3.26}$$

The set of points where $\mathcal{C}_{\mathcal{Q}}\big(\mathbf{Q}\big)$ attains its global minimum is called the *"Fréchet mean,"* while the set of points where it attains its local minimum is called the *"Karcher mean"* or *"Cartan mean,"* [69, 62, 65].

Definition: The ball $\mathcal{B}_r(\mathbf{Q})$ of radius r is said to be *"regular"* if the supremum of the sectional curvatures upon it is less than $\big(\frac{1}{r}\cdot\frac{\pi}{2}\big)^2$, see [89, 66, 54, 65].

Theorem 3.6.1 (Sufficient condition). *Given the constellation of points $\mathcal{Q}$, the Karcher mean $E(\mathcal{Q})$ is unique if $\exists r$ s.t. $\mathcal{B}_r(E(\mathcal{Q}))$ is a normal regular open ball fully containing $\mathcal{Q}$.*

Proof. See [65, 66, 54, 13]. □

Theorem 3.6.2 (Necessary condition). *Consider the following force displacement map:*

$$\mathcal{F}_{\mathcal{Q}}(\mathbf{Q}) \triangleq \mathbf{Q} \dot{+}_{\gamma} \int_{\Psi} \Big(\mathcal{Q}(x) \dot{-}_{\gamma} \mathbf{Q}\Big)\mu(x)dx \tag{3.27}$$

The cost function $\mathcal{C}_{\mathcal{Q}}(\mathbf{Q})$ has a critical point at $\mathbf{Q}$ if and only if $\mathcal{F}_{\mathcal{Q}}(\mathbf{Q}) = \mathbf{Q}$.

Proof. This theorem tells us that the weighted sum of all forces (tangent vectors) at the mean has to be zero. Frequently, these critical points are referred to as *"Riemannian center of mass."* The proof can be found in [89]. □

As far as the computation of expected values is concerned, we emphasize on two methods of computing them. First method is exact, and utilizes some sort of non-linear minimization of the cost function, given in Equation (3.26). This cost function is essentially the variance, which is minimized when the true expected value is found. Moreover, this cost function is convex if we take the normal regular ball of half radius of the one described in Theorem 3.6.1, [65]. This result is due to Karcher. For example, one could consider a Newton method, for which one needs to have some intrinsic manifold computations, [35]. These are the exponential map, the logarithmic map, the parallel transport equation of a tangent vector along a geodesic, and finally the sectional curvature equation. These computations allow to obtain a gradient and a Hessian, see [34], of the cost function given in Equation (3.26) in order to find the expected value. [35] discusses these computations as well as gives an intrinsic Newton algorithm for naturally reductive homogeneous spaces. Other references of interest can be found in [93, 92, 91].

Another approach is approximate. Under certain conditions one can compute an approximate expected value explicitly and linearly, using only the exponential and the logarithmic maps. The computation is performed directly based on Equation (3.23). First, one verifies that Equation (3.23) allows approximate arbitrary parenthesization. Then the recursive computation is performed till each group is left with two points. For a two point barycentric

combination one uses an appropriately chosen point along the minimal geodesic segment connecting these two points. We should note that in spite of the parenthesization chosen, the final value will differ slightly, assuming the constellation of points was sufficiently nice. From the practical application standpoint, this approximation could be sufficient enough. Alternatively, it could be used as an initial guess into the non-linear minimization of the cost function algorithm.

We derive this algorithm for the Grassmann manifold in Section 4.9, and show that the values it produces result in insignificant to most applications bias. The same algorithm might be applicable to other manifolds as well, however it is out of scope of this thesis.

Now, we present the notions developed in the previous sections as they apply to the orthogonal group, which naturally forms a Riemannian manifold. These derivations will be required later in Chapter 4.

3.7 Orthonormal group $\mathcal{O}_n$ as a Riemannian manifold

In this section we study the Lie group of orthogonal transformations. These derivations are well known, and presented here for completeness. For further information we refer the reader to [38, 33].

An orthogonal group $\mathcal{O}_\mathrm{n}$ is given by $\mathcal{O}_\mathrm{n} = \left\{\mathbf{Q} \in \mathbb{R}^n_n : \mathbf{Q}^T\mathbf{Q} = \mathbf{I}\right\}$. It consists of orthonormal square $n \times n$ matrices. Suppose $\mathcal{O}_\mathrm{n}$ is naturally embedded in $\mathbb{R}^n_n$ with a canonical Levi-Civita connection and the intrinsic inner product. Then the reader familiar with the theory of embedded Riemannian manifolds would immediately be able to say that $\mathcal{O}_\mathrm{n}$ is the group with bi-invariant metric. Thus, geodesics are the one-parameter subgroup. However, in order to familiarize the reader with the concepts, we will derive the equation of the geodesic from scratch. Suppose $\gamma(t) = \mathbf{Q}_t$ is a smooth curve in $\mathcal{O}_\mathrm{n}$. By differentiating the orthogonality condition $\mathbf{Q}_t^T\mathbf{Q}_t = \mathbf{I}$ with respect to t twice we get:

$$\mathbf{Q}_t^T\dot{\mathbf{Q}}_t + \dot{\mathbf{Q}}_t^T\mathbf{Q}_t = 0 \tag{3.28}$$

$$\mathbf{Q}_t^T\ddot{\mathbf{Q}}_t + 2\dot{\mathbf{Q}}_t^T\dot{\mathbf{Q}}_t + \ddot{\mathbf{Q}}_t^T\mathbf{Q}_t = 0 \tag{3.29}$$

From Equation (3.28), the set of all derivatives $\boldsymbol{\Delta} = \dot{\mathbf{Q}}$ forms a tangent space at $\mathbf{Q} \in \mathcal{O}_{\mathrm{n}}$:

$$\mathcal{T}_{\mathbf{Q}}(\mathcal{O}_{\mathrm{n}}) = \left\{\boldsymbol{\Delta} : \mathbf{Q}^T\boldsymbol{\Delta} + \boldsymbol{\Delta}^T\mathbf{Q} = 0\right\} \tag{3.30}$$

We define the matrix $K = \mathbf{Q}^T\boldsymbol{\Delta}$. Then $\boldsymbol{\Delta} = \mathbf{Q}K$. By substituting $\boldsymbol{\Delta}$ into the Equation (3.30), we get $\mathbf{Q}^T(\mathbf{Q}K) + (K^T\mathbf{Q}^T)\mathbf{Q} = K + K^T = 0$, so matrix K must be skew-symmetric. Then we can write the tangent space and its orthogonal normal space as:

$$\mathcal{T}_{\mathbf{Q}}(\mathcal{O}_{\mathrm{n}}) = \left\{\boldsymbol{\Delta} : \boldsymbol{\Delta} = \mathbf{Q}K \ , \ K^T + K = 0\right\} \tag{3.31}$$

$$\mathcal{N}_{\mathbf{Q}}(\mathcal{O}_{\mathrm{n}}) = \left\{\boldsymbol{\Delta} : \boldsymbol{\Delta} = \mathbf{Q}S \ , \ S^T = S\right\} \tag{3.32}$$

The canonical Riemannian metric of $\mathcal{O}_{\mathrm{n}}$ coincides with the usual inner product of matrices:

$$\left\langle \boldsymbol{\Delta}, \boldsymbol{\Delta} \right\rangle_{\mathbf{Q}} = \mathrm{trace}\big(\boldsymbol{\Delta}^T\boldsymbol{\Delta}\big) = -\mathrm{trace}\big(K^2\big) \tag{3.33}$$

and after the standard process of polarization of bilinear forms one gets:

$$\left\langle \boldsymbol{\Delta}_1, \boldsymbol{\Delta}_2 \right\rangle = \mathrm{trace}\big(\boldsymbol{\Delta}_1^T\boldsymbol{\Delta}_2\big) \tag{3.34}$$

An equation of the geodesic $\gamma(t) = \mathbf{Q}_t$ is a second order differential equation for which the second derivative lies in the normal space. If we write $\ddot{\mathbf{Q}}_t = \mathbf{Q}_t S \in \mathcal{N}_{\mathbf{Q_t}}(\mathcal{O}_{\mathrm{n}})$ for some symmetric matrix S and plug it into Equation (3.29) we obtain:

$$S + 2\dot{\mathbf{Q}}_t^T\dot{\mathbf{Q}}_t + S^T = 0 \tag{3.35}$$

It follows $S = -\dot{\mathbf{Q}}_t^T\dot{\mathbf{Q}}_t$ and $\ddot{\mathbf{Q}}_t = \mathbf{Q}_t S = -\mathbf{Q}_t(\dot{\mathbf{Q}}_t^T\dot{\mathbf{Q}}_t)$. Then the differential equation of the geodesic $\gamma(t)$ is given by:

$$\ddot{\mathbf{Q}}_t + \mathbf{Q}_t\left(\dot{\mathbf{Q}}_t^T\dot{\mathbf{Q}}_t\right) = 0 \tag{3.36}$$

Another important equation is the equation of the parallel vector field of the tangent vector $\gamma'(0)$ along the curve $\gamma(t)$. But since $\gamma(t)$ is a geodesic, the parallel vector field of the $\gamma'(0)$ will coincide with the geodesic's velocity vector field $\gamma'(t)$, and will be given by $\gamma'(t) = \mathbf{Q}_t\mathbf{Q}_0^T\dot{\mathbf{Q}}_0$. Then it is easily verified that the solution to the initial value problem in

Equation (3.36) for the geodesic vector field for $\mathcal{O}_{\mathrm{n}}$ manifold is given by:

$$\begin{aligned} \gamma(t) &= \mathbf{Q} \cdot e^{Kt} & \gamma(0) &= \mathbf{Q} \\ \gamma'(t) &= \mathbf{Q} \cdot e^{Kt} \cdot K & \gamma'(0) &= \mathbf{Q} \cdot K \end{aligned} \tag{3.37}$$

where K is an arbitrary skew-symmetric matrix. We can obtain the dimensionality of the manifold $\mathcal{O}_{\mathrm{n}}$ from the dimensionality of $\mathcal{T}_{\mathbf{Q}}(\mathcal{O}_{\mathrm{n}})$. The dimensionality is the number of degrees of freedom of skew-symmetric matrices:

$$\dim(\mathcal{O}_{\mathrm{n}}) = \frac{n(n-1)}{2} \tag{3.38}$$

We can also write the geodesic equation using an addition operation of the initial point $\mathbf{Q}$ to the initial tangent $\mathbf{\Delta} = \mathbf{Q}K$:

$$\gamma(t) = \mathbf{Q} \dot{+}_{\gamma} t \cdot \mathbf{Q}K = \mathbf{Q} \cdot (\mathbf{I} \dot{+}_{\gamma} Kt) \triangleq \mathbf{Q} \cdot e^{0+Kt} \tag{3.39}$$

In particular, we will have:

$$\exp_{\mathbf{Q}}\left(\mathbf{Q}K\right) = \mathbf{Q} \dot{+}_{\gamma} \mathbf{Q}K = \mathbf{Q} \cdot e^{K} \tag{3.40}$$

so, one can see that the exponential map, i.e., the addition of a point to a vector, in this particular example is defined via the skew-symmetric matrix exponential. In fact, this is true for all matrix Lie groups, see [8, 38]. The computation of the matrix exponential is straight forward and is described in [79]. We can also express the subtraction of two points that is the logarithmic map as:

$$\log_{\mathbf{Q}}\left(\mathbf{Q} \cdot e^{K}\right) = \mathbf{Q} \cdot e^{K} \dot{-}_{\gamma} \mathbf{Q} = \mathbf{Q} \cdot \left(e^{K} \dot{-}_{\gamma} \mathbf{I}\right) = \mathbf{Q} \cdot K \tag{3.41}$$

so it is clear that subtraction of two points will be given via the matrix logarithm operation. It is well known that the matrix exponential is surjective, so matrix logarithm does exist, however it is not unique, [39]. Moreover, it is complex valued. Note that the orthonormal group is compact, so the logarithmic map exists and is unique for points that lie within the interior of each other, as discussed in Section 3.5. The logarithm corresponding to the value of the logarithmic map is called the *"principal matrix logarithm."* Its computation is not trivial, [28, 53], and is a continuous topic of research.

We should also note that $\mathcal{O}_{\mathrm{n}}$ is a compact manifold and thus is complete, but it is not connected. Its consists of two connected components $\mathcal{SO}_{\mathrm{n}}^{+}$ and $\mathcal{SO}_{\mathrm{n}}^{-}$ consisting of orthonormal matrices of positive, and correspondingly, negative determinant. Thus, points from two different components cannot be joined by a geodesic, [38].

As far as the computation of expected values and barycentric combinations goes, there are a number of papers that discuss the notion of a mean rotation. For example, see [78]. This paper discusses the computation of the Riemannian mean for a constellation of points that lie along the same minimal geodesic segment. They show that for a constellation of n points the mean is the n-th root of their product that yields the minimal value of the cost function (3.26). As one can see, this explicit formula is computationally not very efficient. Moreover, as far as we know, there is no explicit formula for computing the Riemannian mean for the general case, but it can be computed using some form of non-linear iteration [101].

The formulas that we obtained here for the orthonormal group will be used in Chapter 4.

3.8 Summary

In this chapter we addressed background material in the theory of embedded Riemannian manifolds that is necessary for the understanding of the consecutive chapters. We started by explaining the notions of tangent spaces and the tangent bundle of the manifold, vector fields and flows, geodesics and associated geodesic vector fields, the notions of distance. Then we presented the reader with an intuitive representation for the exponential map as an addition of a vector to a point, and its inverse — the logarithmic map — as a subtraction of two points. In addition we gave various definitions for the mean on a Riemannian manifold and defined a notion of barycentric combinations. Finally, we applied these notions on the example of the orthogonal group $\mathcal{O}_{\mathrm{n}}$ when it is considered as an embedded Riemannian manifold.

4. COMPUTATIONS ON THE GRASSMANN MANIFOLD AND AFFINE INVARIANCE

4.1 Preliminaries

In mathematics, a *"Grassmannian"* $\mathcal{G}r_{n,p}(\mathbb{F})$ is the space of all p-dimensional subspaces of an n-dimensional vector space $\mathbb{F}^n$. Alternatively, one can view a Grassmannian to be the space of k-dimensional hyperplanes of $\mathbb{F}^n$ passing through the origin. Therefore it can be thought of as a generalization of projective space. Grassmannian carries a natural geometric structure derived from $\mathbb{F}^n$. In particular, when $\mathbb{F}^n$ is simply $\mathbb{R}^n$, the $\mathcal{G}r_{n,p} \triangleq \mathcal{G}r_{n,p}(\mathbb{R})$ can be given the structure of a smooth manifold embedded in $\mathbb{R}^n$. The $\mathcal{G}r_{n,p}$ is called a real *"Grassmann manifold."*

We remind the reader that in Chapter 2 we introduced an injection that maps $\mathcal{G}A_p$-invariant shapes into the Grassmann manifold. Thus, for affine-invariant shape representation each shape is simply a point of the Grassmann manifold. In this chapter, we focus on efficient geometric algorithms on the Grassmann manifold $\mathcal{G}r_{n,p}$. First, we introduce the numerical representation of points on the Grassmann manifold. Then, we derive formulas that allow computation of basic notions required for most algorithms, such as distance between two points, sample mean of k points, etc. To our best knowledge these formulas are new and have never been published anywhere else, unless we comment otherwise by giving a reference.

4.2 Efficient numerical representation for points on $\mathcal{G}r_{n,p}$

Each point of the Grassmann manifold is a p-dimensional subspace of $\mathbb{R}^n$. Let $\mathbf{p} \in \mathcal{G}r_{n,p}$ and consider an $n \times p$ orthonormal matrix $\mathbf{Y}$ that forms a basis of $\mathbf{p}$. Then clearly $\mathbf{p} = \text{span}(\mathbf{Y})$.

Let us also pick any completion of $\mathbf{Y}$ to form a full basis for $\mathbb{R}^n$. It will be an $n \times (n-p)$ orthonormal matrix $\mathbf{Y}_\perp$ such that $\mathbf{Y}^T\mathbf{Y}_\perp = 0$. We also form a third orthonormal matrix:

$$\underset{n\times n}{\mathbf{Q}} \triangleq \left[\begin{array}{c|c} \underset{n\times p}{\mathbf{Y}} & \underset{n\times n-p}{\mathbf{Y}_\perp} \end{array} \right] \tag{4.1}$$

which is a basis to the full space $\mathbb{R}^n$. Clearly, one can uniquely identify the p-dimensional subspace $\mathbf{p}$ by either one of the three matrices: $\mathbf{Y}$, $\mathbf{Y}_\perp$, or $\mathbf{Q}$. Identification by $\mathbf{Y}$ is trivial. To see in the case of $\mathbf{Y}_\perp$ observe that span($\mathbf{Y}_\perp$) will be an $(n-p)$-dimensional subspace of $\mathbb{R}^n$, perpendicular to $\mathbf{p}$ and complementing it to full vector space $\mathbb{R}^n$. So we can easily recover $\mathbf{p}$ if we were given the $\mathbf{Y}_\perp$ matrix. In fact, one can see that $\mathcal{G}r_{\mathrm{n,p}}$ and $\mathcal{G}r_{\mathrm{n\text{-}p,p}}$ are isomorphic to each other. For this reason, we can always assume that $p \leq \frac{n}{2}$. A similar argument can be applied to the $\mathbf{Q}$ matrix. It carries just enough information to recover $\mathbf{p}$.

We shall note that the representation of $\mathbf{p}$ by either one of $\mathbf{Y}$, $\mathbf{Y}_\perp$, or $\mathbf{Q}$ is surjective but not injective. In other words, there will be more than one $n \times p$ orthonormal matrices $\mathbf{Y}$ that map to the same $\mathbf{p}$. In order to make the mapping injective, we consider left cosets of appropriatly chosen groups with respect to the above matrices. In Chapter 2 we introduced a notion of left cosets a.k.a. equivalency classes or simply orbits. Consider the group $\mathcal{O}_\mathrm{p}$ of all $p \times p$ orthonormal matrices. Then, the left coset of $\mathcal{O}_\mathrm{p}$ with respect to matrix $\mathbf{Y}$ is given by:

$$\lceil \mathbf{Y} \rfloor \triangleq \lceil \mathbf{Y} \rfloor_{\mathcal{O}_\mathrm{p}} = \mathbf{Y}\mathcal{O}_\mathrm{p} = \left\{ \mathbf{Y} \cdot V_p : V_p \in \mathcal{O}_\mathrm{p} \right\} \tag{4.2}$$

In other words, the left coset with respect to $\mathbf{Y}$ consists of all possible orthonormal basis that span the same p-dimensional subspace $\mathbf{p}$. Similarly, we will have

$$\lceil \mathbf{Y}_\perp \rfloor \triangleq \lceil \mathbf{Y}_\perp \rfloor_{\mathcal{O}_{\mathrm{n\text{-}p}}} = \mathbf{Y}_\perp \mathcal{O}_{\mathrm{n\text{-}p}} = \left\{ \mathbf{Y}_\perp \cdot V_{n-p} : V_{n-p} \in \mathcal{O}_{\mathrm{n\text{-}p}} \right\} \tag{4.3}$$

Even though the group under consideration is different in both cases, for simplicity of notation we will simply write $\lceil \mathbf{Y} \rfloor$ instead of $\lceil \mathbf{Y} \rfloor_{\mathcal{O}_\mathrm{p}}$ and $\lceil \mathbf{Y}_\perp \rfloor$ instead of $\lceil \mathbf{Y}_\perp \rfloor_{\mathcal{O}_{\mathrm{n\text{-}p}}}$. The reader should not get confused as the group dimensionality can be simply guessed from the dimensionality of the matrix. We remind that the matrix $\mathbf{Y}$ has p columns, while the matrix $\mathbf{Y}_\perp$ has $(n-p)$ columns. We combine $\lceil \mathbf{Y} \rfloor$ and $\lceil \mathbf{Y}_\perp \rfloor$ together in order to obtain a

similar expression for the left coset of $(\mathcal{O}_{\mathrm{p}} \times \mathcal{O}_{\mathrm{n\text{-}p}})$ with respect to matrix $\mathbf{Q}$. As before, we define the shorthand notation $\lceil \mathbf{Q} \rfloor \triangleq \lceil \mathbf{Q} \rfloor_{(\mathcal{O}_{\mathrm{p}} \times \mathcal{O}_{\mathrm{n\text{-}p}})} = \mathbf{Q}(\mathcal{O}_{\mathrm{p}} \times \mathcal{O}_{\mathrm{n\text{-}p}})$ and write:

$$\lceil \mathbf{Q} \rfloor = \left\{ \mathbf{Q} \cdot \begin{bmatrix} V_p & 0 \\ 0 & V_{n-p} \end{bmatrix} : \begin{array}{c} V_p \in \mathcal{O}_{\mathrm{p}} \\ V_{n-p} \in \mathcal{O}_{\mathrm{n\text{-}p}} \end{array} \right\} = \mathbf{Q} \cdot \begin{bmatrix} \mathcal{O}_{\mathrm{p}} & 0 \\ 0 & \mathcal{O}_{\mathrm{n\text{-}p}} \end{bmatrix} \tag{4.4}$$

This formula shows that we allow a simultaneous change of basis within both the p-dimensional subspace spanned by the first p columns of $\mathbf{Q}$ and the $(n-p)$-dimensional subspace spanned by the last $(n-p)$ columns of $\mathbf{Q}$ without allowing changes to the subspaces themselves. Thus, any representative from the left coset $\lceil \mathbf{Q} \rfloor$ will be identifying the same point $\mathbf{p}$ of the Grassmann manifold.

We now summarize the results in the following lemma.

Lemma 4.2.1. *Either one of* $\lceil \mathbf{Y} \rfloor$ *or* $\lceil \mathbf{Y}_\perp \rfloor$ *or* $\lceil \mathbf{Q} \rfloor$ *left cosets are bijective representations of the point* $\mathbf{p}$ *on the Grassmann manifold* $\mathcal{G}r_{\mathrm{n,p}}$*, and we simply write* $\mathbf{p} = \lceil \mathbf{Y} \rfloor = \lceil \mathbf{Y}_\perp \rfloor = \lceil \mathbf{Q} \rfloor$.

Proof. Proof is trivial and well known. We refer the reader to [38] for further information. We just note that these representations arise naturally from the construction of the Grassmann manifold as a quotient of various well known Lie groups. In particular, in order to get the $\lceil \mathbf{Y} \rfloor$ representation, one needs to consider $\mathcal{G}r_{\mathrm{n,p}} = \mathcal{G}f_{\mathrm{n,p}}/\mathcal{O}_{\mathrm{p}}$. The $\lceil \mathbf{Y}_\perp \rfloor$ representation comes from the fact that $\mathcal{G}r_{\mathrm{n,p}}$ and $\mathcal{G}r_{\mathrm{n\text{-}p,p}}$ are isomorphic to each other. Finally, the $\lceil \mathbf{Q} \rfloor$ representation is derived by considering $\mathcal{G}r_{\mathrm{n,p}} = \mathcal{O}_{\mathrm{n}}/(\mathcal{O}_{\mathrm{p}} \times \mathcal{O}_{\mathrm{n\text{-}p}})$ □

Throughout the chapter, we will use all three notations for representing points on the Grassmann manifold. Depending on a particular scenario, we may need to use one or another representation in order to derive certain formulas. However, in order to represent points of the Grassmann manifold numerically, we follow the approach given in [33]. We store points numerically using any $n \times p$ matrix $\hat{\mathbf{Y}} \in \lceil \mathbf{Y} \rfloor$, and we give final formulas in the form that will operate on these matrices. For most practical applications, p is generally very small (for example, for 3D shapes $p = 3$), so this representation is very efficient.

As a side note, we should mention that these representations are not the only possible choices. For example, one could use Grassmann (Plücker) coordinates, see [112, 71, 55, 74], or projection matrices for subspaces, see [27, 51, 88]. Especially interesting are the results presented in [27], where the Grassmannian $\mathcal{G}r_{\mathrm{n,p}}$ equipped with the chordal metric, which

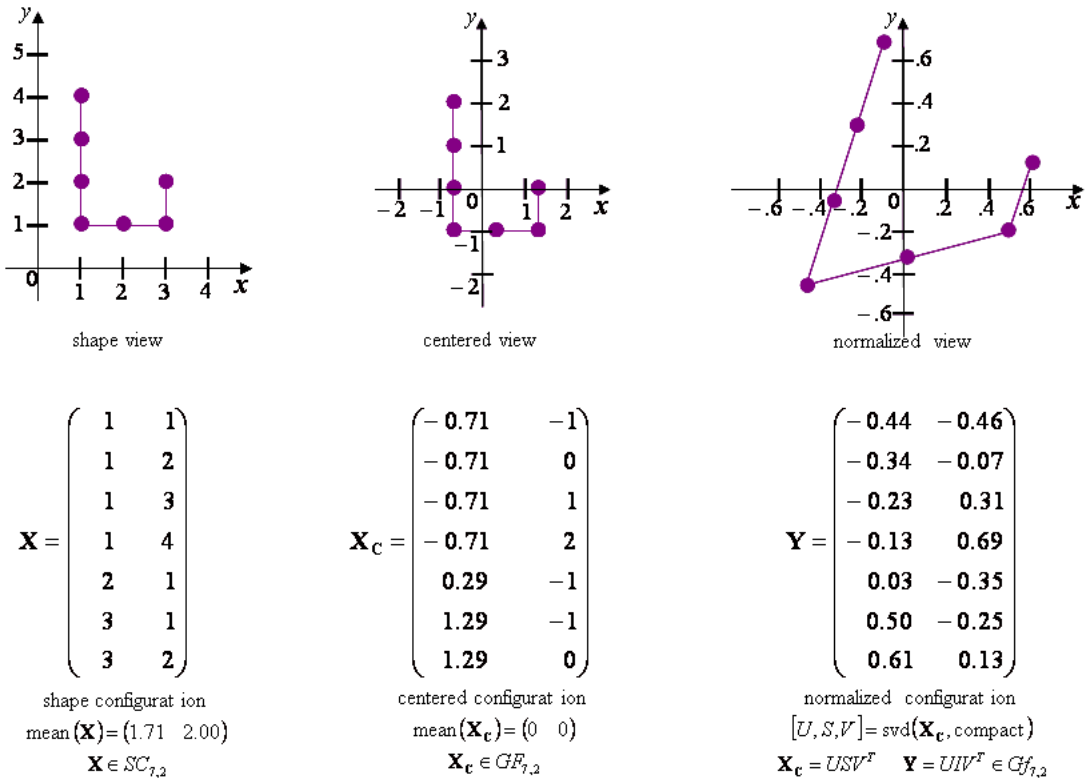

Fig. 4.1: Obtaining a canonical representation of the shape 'L' for storage

is the straight-line Euclidean distance in between the projection matrices, is shown to be isometrically embedded in $\mathbb{R}^d$ where $d = \frac{n(n+1)}{2} - 1$ and does not depend on p. Moreover, it is shown that a family of all $\mathcal{G}r_{\mathrm{n,p}}$ with n being fixed and p taking all possible values, lie on a hyper-sphere in $\mathbb{R}^d \times \mathbb{R}^1$ of radius $\frac{\sqrt{n}}{2}$. Representations by (Plücker) coordinates and by projection matrices are either not memory efficient or hard to work with for deriving efficient algorithms. We do not use them in this work.

Now we briefly summarize one more time the connection between affine-invariant shapes and points on the Grassmann manifold and the canonical representation of shapes in memory for storage and retrieval requirements by going through a quick example. In this example we consider a 2D shape for the capital letter 'L'. Consider a view of this shape given on the left in Figure 4.1. This shape is composed of seven particles. Each particle has (x, y) coordinates. Also, in this example we assume that particles are labeled 1 through 7. We

stack the coordinates of the particles according to the ascending order of their labels into the matrix $\mathbf{X}$ as shown on the left of Figure 4.1. As a side note, if particles were not labeled, then we would stack their coordinates into matrix $\mathbf{X}$ in any arbitrarily picked order. As one may recall, this matrix $\mathbf{X}$ belongs to the set of configurations $\mathcal{SC}_{n,p}$, shown in the third block of the diagram of Figure 2.2. We perform three pre-processing steps on the 7×2 matrix $\mathbf{X}$. The first step is removing the translation ambiguity by shifting the coordinate axes to the center of mass of the shape view. The center of mass is found by calculating the column-wise mean of $\mathbf{X}$. One finds that the column-wise mean$(\mathbf{X}) = \big(1.71, 2.00\big)$ in this example. We subtract it from each row in order to obtain a centered configuration $\mathbf{X_c}$ and associated with it a centered view, as shown in the middle of Figure 4.1. $\mathbf{X_c}$ obtained in this way belongs to the set of centered configurations $\mathcal{SM}_{n,p}$, as depicted in the fourth block in the bottom left corner of Figure 2.2. The second step is a simple check to verify that $\mathbf{X_c}$ is rank-2 matrix. Once the shape passes this check, it is a point of the non-compact Stiefel manifold $\mathcal{GF}_{(7,2)}$, as shown in the fifth block of Figure 2.2. Note that only shape views that have particles located along a straight line fail this check. Also, 2D shape views whose particles are located almost along a straight line will be rank-2, but will have a really bad condition number. Essentially, views of such shapes are so badly distorted by a 2D affine transformation that the information about the shape cannot be reliably recovered from these views. Such views should also be omitted. The third and final step is a projection onto the Stiefel manifold $\mathcal{G}f_{7,2}$. It is obtained by means of a compact SVD decomposition. First, we decompose the matrix $\mathbf{X_c} = USV^T$ via a compact SVD. Then we replace the singular values with ones and construct an orthonormal 7×2 matrix $\mathbf{Y} = UIV^T$, as shown on the right of Figure 4.1. One notes that the obtained orthonormal matrix $\mathbf{Y}$ still has a zero column-wise mean. This 7×2 matrix $\mathbf{Y}$ is the representation of the shape 'L'. It is used for storage and retrieval, and is an input to the formulas that will be presented further in this dissertation. Shape itself will be a left coset $\lceil \mathbf{Y} \rfloor \in \mathcal{G}r_{\mathrm{n,p}}$, which is a point of the Grassmann manifold, as we already discussed. One should note that the obtained matrix $\mathbf{Y}$ is just one of the possible representations of the shape. We are not obtaining a canonical

representative of the left coset $\lceil \mathbf{Y} \rfloor$, as opposed to the method in [45]. Instead, we are going to derive formulas that are invariant to choice of a representative of the left coset $\lceil \mathbf{Y} \rfloor$.

4.3 Representing Grassmann manifold $\mathcal{G}r_{n,p}$ as a quotient manifold

In this section we give a representation of the Grassmann manifold as a quotient of lie groups $\mathcal{G}r_{\mathrm{n,p}} = \mathcal{O}_{\mathrm{n}}/(\mathcal{O}_{\mathrm{p}} \times \mathcal{O}_{\mathrm{n\text{-}p}})$. These derivations are well known, and presented here for completeness. For further information we refer the reader to [38, 33, 1, 110, 113, 52, 96].

We start by picking some orthonormal $n \times n$ matrix $\mathbf{Q} \in \mathcal{O}_{\mathrm{n}}$ and its tangent $\mathbf{\Delta} \in \mathcal{T}_{\mathbf{Q}}(\mathcal{O}_{\mathrm{n}})$ as was presented in Section 3.7. Also recall that we showed that matrix $K = \mathbf{Q}^T\mathbf{\Delta}$ is skew-symmetric.

Now we fix $p \leq \frac{n}{2}$. Recall from Section 4.2 why this is possible. We decompose $\mathbf{Q}$, $\mathbf{\Delta}$, and the skew-symmetric matrix K into block matrices as follows:

$$\underset{n\times n}{\mathbf{Q}} = \begin{bmatrix} \underset{n\times p}{\mathbf{Y}} & \underset{n\times n-p}{\mathbf{Y}_\perp} \end{bmatrix} \tag{4.5}$$

$$\underset{n\times n}{\mathbf{\Delta}} = \begin{bmatrix} \underset{n\times p}{\mathbf{H}} & \underset{n\times n-p}{\mathbf{H}_\perp} \end{bmatrix} \tag{4.6}$$

$$\underset{n\times n}{K} = \begin{bmatrix} \underset{p\times p}{F} & \underset{p\times n-p}{-R^T} \\ \underset{n-p\times p}{R} & \underset{n-p\times n-p}{G} \end{bmatrix} \tag{4.7}$$

where F and G are skew-symmetric and R is an arbitrary matrix. Now we can rewrite a general form of a tangent $\mathbf{\Delta} = \mathbf{Q}K$ as:

$$\mathbf{\Delta} = \left[\begin{array}{c|c} \mathbf{H} & \mathbf{H}_\perp \end{array}\right] = \left[\begin{array}{c|c} \mathbf{Y}F + \mathbf{Y}_\perp R & -\mathbf{Y}R^T + \mathbf{Y}_\perp G \end{array}\right] \tag{4.8}$$

The dimensionality of the manifold $\mathcal{G}r_{\mathrm{n,p}} = \mathcal{O}_{\mathrm{n}}/\mathcal{O}_{\mathrm{p}} \times \mathcal{O}_{\mathrm{n\text{-}p}}$ is given by:

$$\dim(\mathcal{G}r_{\mathrm{n,p}}) = \dim(\mathcal{O}_{\mathrm{n}}) - \dim(\mathcal{O}_{\mathrm{p}}) - \dim(\mathcal{O}_{\mathrm{n\text{-}p}}) = p(n-p) \tag{4.9}$$

A point $\lceil \mathbf{Q} \rfloor \in \mathcal{G}r_{\mathrm{n,p}} \equiv \mathcal{O}_{\mathrm{n}}/\mathcal{O}_{\mathrm{p}} \times \mathcal{O}_{\mathrm{n\text{-}p}}$ on the Grassmann manifold is given by Equation (4.4).

The tangent space $\mathcal{T}_{\mathbf{Q}}(\mathcal{O}_{\mathrm{n}})$ at $\mathbf{Q} \in \mathcal{O}_{\mathrm{n}}$, as given by Equation (3.31), will also get split into two components, the vertical component $\mathcal{V}_{\mathbf{Q}}(\mathcal{O}_{\mathrm{n}})$ and the horizontal component $\mathcal{H}_{\mathbf{Q}}(\mathcal{O}_{\mathrm{n}})$. The vertical component is defined such that by taking any vertical tangent and making a

move along the geodesic in $\mathcal{O}_{\mathrm{n}}$ produces no move in the quotient space $\mathcal{G}r_{\mathrm{n,p}} = \mathcal{O}_{\mathrm{n}}/\mathcal{O}_{\mathrm{p}} \times \mathcal{O}_{\mathrm{n\text{-}p}}$. From Equation (4.8), it is easy to see that:

$$\boldsymbol{\Delta}_V = \left[\begin{array}{c|c} \mathbf{Y}F & \mathbf{Y}_\perp G \end{array} \right] = \mathbf{Q} \cdot \begin{bmatrix} F & 0 \\ 0 & G \end{bmatrix} \tag{4.10}$$

$$\boldsymbol{\Delta}_H = \left[\begin{array}{c|c} \mathbf{Y}_\perp R & -\mathbf{Y}R^T \end{array} \right] = \mathbf{Q} \cdot \begin{bmatrix} 0 & -R^T \\ R & 0 \end{bmatrix} \tag{4.11}$$

The fact that a move from point $\mathbf{Q} = \mathbf{Q}_0 \in \lceil \mathbf{Q} \rfloor$ along the geodesic in $\mathcal{O}_{\mathrm{n}}$ in the direction of a vertical tangent will move us to point $\mathbf{Q}_\tau \in \lceil \mathbf{Q} \rfloor$, which is still the same point in the quotient space $\lceil \mathbf{Q} \rfloor \in \mathcal{G}r_{\mathrm{n,p}}$, can be verified by plugging $\boldsymbol{\Delta}_V$ from Equation (4.10) into Equation (3.37):

$$\mathbf{Q}_\tau = \mathbf{Q}_0 \cdot \exp\left(\begin{bmatrix} F & 0 \\ 0 & G \end{bmatrix} t \right) = \mathbf{Q}_0 \cdot \begin{bmatrix} e^{Ft} & 0 \\ 0 & e^{Gt} \end{bmatrix} = \mathbf{Q}_0 \cdot \begin{bmatrix} V_p & 0 \\ 0 & V_{n-p} \end{bmatrix} \in \lceil \mathbf{Q} \rfloor \tag{4.12}$$

The tangent space of a point on the Grassmann manifold is in bijective correspondence with the horizontal component $\mathcal{T}_{\lceil \mathbf{Y} \rfloor}(\mathcal{G}r_{\mathrm{n,p}}) \cong \mathcal{T}_{\lceil \mathbf{Q} \rfloor}(\mathcal{G}r_{\mathrm{n,p}}) \cong \mathcal{H}_{\mathbf{Q}}(\mathcal{O}_{\mathrm{n}})$ under the natural projection. Thus, tangent spaces for the $\mathcal{G}r_{\mathrm{n,p}}$ are of the form:

$$\mathcal{T}_{\lceil \mathbf{Q} \rfloor}(\mathcal{G}r_{\mathrm{n,p}}) = \left\{ \lceil \boldsymbol{\Delta} \rfloor : \lceil \boldsymbol{\Delta} \rfloor = \lceil \mathbf{Q} \rfloor \cdot \begin{bmatrix} 0 & -R^T \\ R & 0 \end{bmatrix} \right\} \tag{4.13}$$

where R is an arbitrary $(n-p) \times p$ matrix. In fact, one can immediately verify that the dimensionality of the Grassmann manifold $\mathcal{G}r_{\mathrm{n,p}}$, which is the same as the dimensionality of its tangent space $\mathcal{T}_{\lceil \mathbf{Q} \rfloor}(\mathcal{G}r_{\mathrm{n,p}})$, is the $p(n-p)$ degrees of freedom of the matrix R. Equation (4.13) is the general form of the tangent space. In particular, when we study the Grassmann manifold by working only with the subspace spanned by the first p-columns given by $\mathbf{Y}$, we can identify the tangent space by:

$$\mathcal{T}_{\lceil \mathbf{Y} \rfloor}(\mathcal{G}r_{\mathrm{n,p}}) = \left\{ \lceil \mathbf{H} \rfloor : \lceil \mathbf{H} \rfloor = \lceil \mathbf{Y}_\perp \rfloor \cdot R \right\} \tag{4.14}$$

The canonical Riemannian metric on the Grassmann manifold when it is considered as an embedding in the Stiefel manifold, i.e., $\mathcal{G}r_{\mathrm{n,p}} = \mathcal{G}f_{\mathrm{n,p}}/\mathcal{O}_{\mathrm{p}}$, is given by:

$$\Big\langle \lceil \mathbf{H} \rfloor, \lceil \mathbf{H} \rfloor \Big\rangle_{\lceil \mathbf{Y} \rfloor} = \mathrm{trace}\big(\mathbf{H}^T \mathbf{H}\big) = \mathrm{trace}\big(R^T R\big) \tag{4.15}$$

Thus, it coincides with the canonical Euclidean metric of $\mathbb{R}^n_p$. Similarly, when $\mathcal{G}r_{\mathrm{n,p}}$ is considered as an embedding in the orthogonal group, i.e., $\mathcal{G}r_{\mathrm{n,p}} = \mathcal{O}_{\mathrm{n}}/(\mathcal{O}_{\mathrm{p}} \times \mathcal{O}_{\mathrm{n\text{-}p}})$, its canonical metric coincides with the metric of $\mathcal{O}_{\mathrm{n}}$ up to multiplication by $\frac{1}{2}$ and is given by:

$$\Big\langle \lceil \boldsymbol{\Delta} \rfloor, \lceil \boldsymbol{\Delta} \rfloor \Big\rangle_{\lceil \mathbf{Q} \rfloor} = \frac{1}{2}\mathrm{trace}\big(\boldsymbol{\Delta}^T \boldsymbol{\Delta}\big) = -\frac{1}{2}\mathrm{trace}\big(K^2\big) \tag{4.16}$$

where the matrix $\boldsymbol{\Delta} = \mathbf{Q}K$, see Equation (3.31), with K skew-symmetric. Moreover, $\boldsymbol{\Delta}$ is the restriction to the horizontal space, i.e., it is of the form given by Equation (4.11). Since $\mathrm{trace}\big(R^T R\big) = -\frac{1}{2}\mathrm{trace}\big(K^2\big)$ for horizontal tangents, one immediately verifies that the two definitions of the canonical metric coincide.

Equation (3.37) of the geodesic and of its velocity vector field in $\mathcal{O}_{\mathrm{n}}$ will also be the equation of the geodesic and of its velocity vector field in $\mathcal{G}r_{\mathrm{n,p}}$ where we take tangents of the form given by Equation (4.11). But these equations can be written in more computationally efficient form, when we are dealing with tangents of the form in Equation (4.14).

4.4 Linear solution for the geodesic flow — the exponential map

Edelman, Arias, and Smith [33] have proposed a computationally practical way of defining equations for the geodesic and its geodesic vector field, given an initial point and a tangent direction at the point. More general equations are given in [1]. This is known as an *initial value problem* in the theory of differential equations. Their approach does not require an explicit formula for the connection. For completeness, we present their result below.

Consider a point $\lceil \mathbf{Y}_0 \rfloor \in \mathcal{G}r_{\mathrm{n,p}}$ on the Grassmann manifold and let $\mathbf{Y}_0 \in \lceil \mathbf{Y}_0 \rfloor$ be a representative of the orbit. Also take $\mathbf{H}_0$ to be a valid tangent vector at $\mathbf{Y}_0$. Then according to the Theorem 3.3.2 there exists a unique geodesic γ given by:

$$\dot{\gamma}\big|_{\gamma} = \Pi\Big(\lceil \mathbf{Y}_0 \rfloor, \lceil \mathbf{H}_0 \rfloor\Big)\Big|_{\gamma} \quad : \quad \dot{\gamma}\Big|_{\gamma}(0) = \Big(\gamma(0), \gamma'(0)\Big) = \Big(\lceil \mathbf{Y}_0 \rfloor, \lceil \mathbf{H}_0 \rfloor\Big) \tag{4.17}$$

We perform an SVD decomposition of $\mathbf{H}_0 = UEV^T$. We define matrices $C_t = \mathrm{cosm}(t \cdot E)$ and $S_t = \mathrm{sinm}(t \cdot E)$ to be the matrix-cosine and matrix-sine, which are defined well for symmetric matrices, [76]. In case of the diagonal matrix E, these will also be the diagonal

matrices with diagonal elements that are the cosine and sine of the corresponding elements in E. The result is given by the lemma:

Theorem 4.4.1. *The equation for the geodesic velocity vector field $\dot{\gamma}\big|_{\gamma}$ at time t with the initial values at time $t = 0$ is given by:*

$$\begin{aligned} \gamma(t) = \lceil \mathbf{Y}_t \rfloor &= \lceil \mathbf{Y}_0 \rfloor \cdot V C_t V^T + \lceil \mathbf{H}_0 \rfloor \cdot V S_t E^{-1} V^T \qquad \gamma(0) = \lceil \mathbf{Y}_0 \rfloor \\ \gamma'(t) = \lceil \mathbf{H}_t \rfloor &= -\lceil \mathbf{Y}_0 \rfloor \cdot V S_t E V^T + \lceil \mathbf{H}_0 \rfloor \cdot V C_t V^T \qquad \gamma'(0) = \lceil \mathbf{H}_0 \rfloor \end{aligned} \tag{4.18}$$

Proof. This result is derived in [33]. We give it here for completeness. Note that $p \leq (n-p)$ and let us decompose R from Equation (4.8) using an SVD decomposition as follows:

$$\underset{n-p\times p}{R} = \begin{bmatrix} \underset{n-p\times p}{U} & \underset{n-p\times n-2p}{W} \end{bmatrix} \cdot \begin{bmatrix} \underset{p\times p}{E} \\ \underset{n-2p\times p}{0} \end{bmatrix} \cdot \underset{p\times p}{V}^T = U \cdot E \cdot V^T \tag{4.19}$$

Given this SVD decomposition, it is easily verified that:

$$K = \begin{bmatrix} 0 & -R^T \\ R & 0 \end{bmatrix} = \left[\begin{array}{c|cc} V & 0 & 0 \\ \hline 0 & U & W \end{array}\right] \cdot \left[\begin{array}{c|cc} 0 & -E & 0 \\ \hline E & 0 & 0 \\ 0 & 0 & 0 \end{array}\right] \cdot \left[\begin{array}{c|cc} V & 0 & 0 \\ \hline 0 & U & W \end{array}\right]^T \tag{4.20}$$

Using the fact that $e^{AXA^{-1}} = Ae^X A^{-1}$, we can write:

$$e^{Kt} = \left[\begin{array}{c|cc} V & 0 & 0 \\ \hline 0 & U & W \end{array}\right] \cdot \left[\begin{array}{c|cc} C_t & -S_t & 0 \\ \hline S_t & C_t & 0 \\ 0 & 0 & I \end{array}\right] \cdot \left[\begin{array}{c|cc} V & 0 & 0 \\ \hline 0 & U & W \end{array}\right]^T \tag{4.21}$$

where $C_t = \mathrm{cosm}(Et)$ and $S_t = \mathrm{sinm}(Et)$ are matrix-cosine and matrix-sine. Then:

$$\mathbf{Q}_0 e^{Kt} = \left[\, \mathbf{Y}_0 V C_t V^T + \mathbf{Y}_{\perp 0} U S_t V^T \,\middle|\, -\mathbf{Y}_0 V S_t U^T + \mathbf{Y}_{\perp 0} U C_t U^T + \mathbf{Y}_{\perp 0} W W^T \,\right] \tag{4.22}$$

But $\mathbf{H}_0 = \mathbf{Y}_{\perp 0} R = \mathbf{Y}_{\perp 0} U E V^T$, so we can write $\mathbf{Y}_{\perp 0} U = \mathbf{H}_0 V E^{-1}$. Also, we note that $U^T W = 0$. Then Equation (4.22) becomes:

$$\mathbf{Q}_0 e^{Kt} = \left[\, \mathbf{Y}_0 V C_t V^T + \mathbf{H}_0 V E^{-1} S_t V^T \,\middle|\, -\mathbf{Y}_0 V S_t U^T + \mathbf{H}_0 V E^{-1} C_t U^T \,\right] \tag{4.23}$$

Post-multiplying by K and substituting $U^T R = E V^T$ we get:

$$\mathbf{Q}_0 e^{Kt} K = \left[\, -\mathbf{Y}_0 V S_t E V^T + \mathbf{H}_0 V E^{-1} C_t E V^T \,\middle|\, -\mathbf{Y}_0 V C_t V^T R^T - \mathbf{H}_0 V E^{-1} S_t V^T R^T \,\right] \tag{4.24}$$

Finally, from Equations (3.37), (4.23), (4.24) we obtain the equations for the geodesic vector field by constraining ourself to working just with the subspace spanned by the first p-columns given by $\mathbf{Y}_0$, and by going from $\mathbf{Y}_0$ to its orbit $\lceil \mathbf{Y}_0 \rfloor$. □

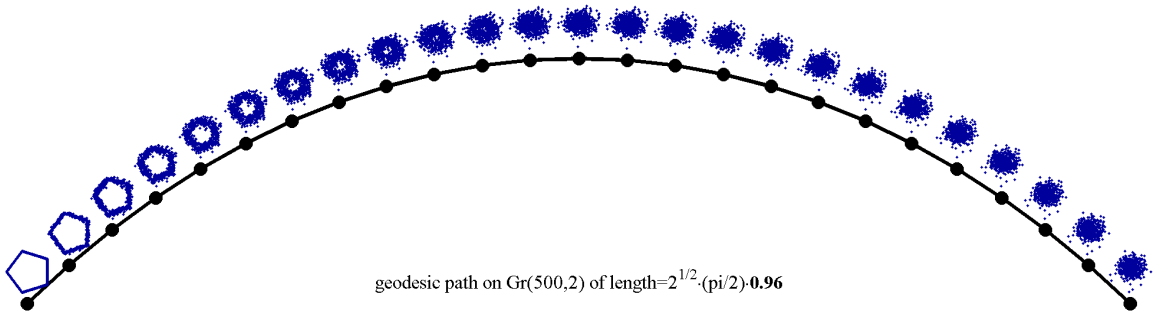

Fig. 4.2: A geodesic flow in a random direction

Recall from Section 3.5 that the Equation (4.18) gives us a solution for the exponential map, i.e. an addition of a vector to a point. We can also compute distances. Suppose we have a geodesic γ defined via the initial value Equation (4.18) of the geodesic. Recall that Theorem 3.4.2 gives us a general form of computing distances along this geodesic. Let $\gamma(t_1)$ and $\gamma(t_2)$ be these two points lying sufficiently close to each other (to be more exact, within the interior of each other, as discussed in Section 3.5). Then we have:

$$\mathrm{d}\Big(\gamma(t_1),\gamma(t_2)\Big) = \Big\|\lceil \mathbf{H}_0 \rfloor\Big\| \cdot \big|t_1 - t_2\big| \tag{4.25}$$

In fact, one can easily verify the theorem by verifying the fact that the norm of $\lceil \mathbf{H}_t \rfloor$ is constant and is equal to:

$$\Big\|\lceil \mathbf{H}_t \rfloor\Big\|_{\gamma(t)} = \Big\|\lceil \mathbf{H}_0 \rfloor\Big\|_{\gamma(0)} = \sqrt{\mathrm{trace}\big(\mathbf{H}_0^T \mathbf{H}_0\big)} = \sqrt{\mathrm{trace}\big(R^T R\big)} = \sqrt{\mathrm{trace}\big(E^2\big)} \tag{4.26}$$

So the distance on the $\mathcal{G}r_{\mathrm{np}}$ manifold along the geodesic from Equation (4.18) is simply:

$$\mathrm{d}\Big(\gamma(t_1),\gamma(t_2)\Big) = \sqrt{\mathrm{trace}(E^2)} \cdot \big|t_1 - t_2\big| \tag{4.27}$$

This equation only tells us how to compute distances along a predefined geodesic. It does not tell us how to compute distance between two arbitrary points on the $\mathcal{G}r_{\mathrm{np}}$. The general formula will be given later on in this chapter.

Now we illustrate the geodesic flow from Equation (4.18) by example. In Figure 4.2 we present a geodesic flow defined by the initial shape $\lceil \mathbf{Y}_0 \rfloor \in \mathcal{G}r_{500,2}$ and the tangent

direction $\lceil \mathbf{H}_0 \rfloor$. In this example, we take our initial shape to be a contour of a pentagon. We pick a tangent direction $\lceil \mathbf{H}_0 \rfloor$ randomly from all tangent vectors at $\lceil \mathbf{Y}_0 \rfloor$. Then, we start moving on the geodesic defined by Equation (4.18), and compute intermediate shapes at a distance $d = \sqrt{2}\frac{\pi}{2}0.04$ from each other for a total of 25 equally spaced shapes along the same geodesic. Note that all these shapes are also defined by 500 points as the original pentagon. The shapes we obtain are not likely to be shapes that we would see in real life. This shows that there are lots of shapes in the shape space that have a very small a-priori probability of happening in the real world. Also, this shows that the tangent vectors that are likely to be followed do have a certain structure. Thus, the problem we would like to solve is quite different from the solution given by Equation (4.18). Instead of defining the equation for the geodesic velocity vector field given an initial point and a tangent direction at the point, it is much more important in applications to define the equation for the geodesic segment connecting an initial point and a final point on $\mathcal{G}r_{\mathrm{n,p}}$. In other words, we are interested in finding a solution for the logarithmic map on the Grassmann manifold $\mathcal{G}r_{\mathrm{n,p}}$. The result is presented in Section 4.8, and the preliminaries are given in the following Sections.

4.5 Principal angles, cut locus, and bounds on radii of regular balls

The Grassmann manifold $\mathcal{G}r_{\mathrm{n,p}}$ is a compact globally symmetric manifold, [16, 52]. Recall from Section 3.5 that compactness guarantees (geodesic) completeness. So every two points $\lceil \mathbf{Y}_0 \rfloor, \lceil \mathbf{Y}_\tau \rfloor \in \mathcal{G}r_{\mathrm{n,p}}$ can be joined by a geodesic segment of length $\mathrm{d}\big(\lceil \mathbf{Y}_0 \rfloor, \lceil \mathbf{Y}_\tau \rfloor\big)$. Moreover, if they lie within the interior of each other, then the geodesic segment will also be unique. For this reason, we first give in this section the properties of the cut locus and the interior on the Grassmann manifold. Then in the subsequent sections we solve for the logarithmic map $\lceil \mathbf{H}_0 \rfloor = \lceil \mathbf{Y}_\tau \rfloor \dot{-}_\wr \lceil \mathbf{Y}_0 \rfloor$ such that $\lceil \mathbf{Y}_0 \rfloor \dot{+}_\wr \lceil \mathbf{H}_0 \rfloor = \lceil \mathbf{Y}_\tau \rfloor$. The solution is broken down in a few steps. In the first step, we define a barycentric combination of two points. We give a formula for computing an equal weight barycentric combination of two points, a.k.a a computation of the midpoint in between two points. Simultaneously, we solve for the logarithmic map at the midpoint. Combining this result with the solution of the exponential

map we give a general form for the logarithmic map. It allows us to give a solution for the geodesic segment connecting two points. This is the two-point boundary value problem of the geodesic. Finally, we give a general form for the barycentric combination of two and more points. The barycentric combination allows us to define the mean of a family of points and to study higher-order statistics in the tangent space attached at the mean.

Definition: Let $\lceil \mathbf{Y}_0 \rfloor, \lceil \mathbf{Y}_\tau \rfloor \in \mathcal{G}r_{\mathrm{n,p}}$ be two points. Recall that each point on the Grassmann manifold is a p-dimensional subspace of $\mathbb{R}^n$. Then the stationary values in between the subspaces are called the *"principal angles"* or the *"canonical angles"* in between these two subspaces, see [15, 40, 75].

The notion of canonical angles between subspaces goes back to Jordan, see [59, 90, 26]. Since each point is a p-dimensional subspace of $\mathbb{R}^n$, there will be n angles of which at most p can be nonzero. The cosines of the principal angles between the column spaces of orthonormal matrices are efficiently computed using the Björck-Golub algorithm, [15, 40]. Consider an SVD decomposition:

$$\mathbf{Y_0}^T \mathbf{Y}_\tau = V_0 C_\tau V_\tau^T \tag{4.28}$$

Then the diagonal matrix C_τ will contain p cosines of the principal angles on its diagonal. To find the principal angles compute $E_\tau = \mathrm{acosm}(C_\tau)$, where $\mathrm{acosm}(\cdot)$ is a matrix arc-cosine. For diagonal matrices, it is given by computing arc-cosines of elements along the diagonal. This is a straightforward cosine based algorithm given in [15, 40]. However, this algorithm cannot provide accurate results for small angles because of the presence of round-off errors. We highly recommend using a more stable implementation. For example, see [67, 31] for combined sine-cosine based algorithms. Now, we study the cut locus. Good references on this topic are [95, 107, 108, 63].

Lemma 4.5.1. *Let $\lceil \mathbf{Y}_0 \rfloor, \lceil \mathbf{Y}_\tau \rfloor \in \mathcal{G}r_{n,p}$ be two points on the Grassmann manifold and let $(\phi_i) : i = 1 \dots p$ be the principal angles in between two points. Then the length of a geodesic*

segment γ joining them is given by:

$$L(\gamma)\Big|_{\lceil \mathbf{Y}_0 \rfloor}^{\lceil \mathbf{Y}_\tau \rfloor} = \Big(\sum_{i=1}^{p} \phi_i^2 \Big)^{\frac{1}{2}} \tag{4.29}$$

Moreover, if each of the angles $\phi_i < \frac{\pi}{2}$, then the geodesic segment will be the unique geodesic segment of shortest length $d\big(\lceil \mathbf{Y}_0 \rfloor, \lceil \mathbf{Y}_\tau \rfloor\big)$.

Proof. See [107]. □

Theorem 4.5.2 (Distance between two points). *Let $\lceil \mathbf{Y}_0 \rfloor, \lceil \mathbf{Y}_\tau \rfloor \in \mathcal{G}r_{n,p}$ be two points on the Grassmann manifold. Then distance in between them is given by:*

$$d\Big(\lceil \mathbf{Y}_0 \rfloor, \lceil \mathbf{Y}_\tau \rfloor\Big) = \sqrt{trace(E_\tau^2)} \tag{4.30}$$

where $E_\tau = acosm(C_\tau)$ is a matrix from the SVD decomposition given in Equation (4.28).

Proof. Proof is trivial, and is a direct consequence of Lemma 4.5.1. The principal angles in the matrix $E_\tau = \text{acosm}(C_\tau)$ will automatically be less than or equal to $\frac{\pi}{2}$. To see this one can observe that the diagonal entries of C_τ are the singular values and will always be non-negative. In fact, one can show that they will be in the range $[0, 1]$ since both $\mathbf{Y_0}$ and $\mathbf{Y}_\tau$ are orthonormal. At the same time, $\text{acos} : [-1, 1] \to [0, \pi]$ is a monotonically decreasing function with $\text{acos}(0) = \frac{\pi}{2}$ and $\text{acos}(1) = 0$. □

Based on proof of Theorem 4.5.2 and Lemma 4.5.1, one notes that the only singularity is achieved and the geodesic segment will not be unique when one of the principal angles is exactly $\frac{\pi}{2}$. We summarize the result in the following theorem.

Theorem 4.5.3 (cut locus theorem). *Let $\lceil \mathbf{Y}_0 \rfloor, \lceil \mathbf{Y}_\tau \rfloor \in \mathcal{G}r_{n,p}$ be two points. $\lceil \mathbf{Y}_0 \rfloor$ is in the cut locus of $\lceil \mathbf{Y}_\tau \rfloor$ and $\lceil \mathbf{Y}_\tau \rfloor$ is in the cut locus of $\lceil \mathbf{Y}_0 \rfloor$ if and only if at least one of the principal angles in between them is equal to $\frac{\pi}{2}$.*

Proof. See [107]. □

Observe that as a consequence of this theorem one can easily see that the maximal distance will be achieved when all principal angles are $\frac{\pi}{2}$, so for any $\lceil \mathbf{Y}_1 \rfloor, \lceil \mathbf{Y}_2 \rfloor \in \mathcal{G}r_{\mathrm{n,p}}$ we have:

$$0 \leq \mathrm{d}\Big(\lceil \mathbf{Y}_1 \rfloor, \lceil \mathbf{Y}_2 \rfloor\Big) \leq \frac{\pi}{2}\sqrt{p} \tag{4.31}$$

For practical purposes it is convenient to normalize the distance so that it is in the range $[0, 1]$. We define the normalized distance below.

Definition: Normalized distance on the Grassmann manifold $\mathcal{G}r_{\mathfrak{n},\mathfrak{p}}$ is defined to be:

$$\mathrm{d}_n\Big(\lceil \mathbf{Y}_0 \rfloor, \lceil \mathbf{Y}_\tau \rfloor\Big) = \frac{1}{\frac{\pi}{2}\sqrt{p}} \cdot \Big(\sum_{i=1}^{p} \phi_i^2\Big)^{\frac{1}{2}} = \frac{1}{\frac{\pi}{2}\sqrt{p}} \cdot \sqrt{\mathrm{trace}(E_\tau^2)} \tag{4.32}$$

where ϕ_i are the principal angles in between two points. It takes values in the range $[0, 1]$.

Recall from Section 3.5 that the interior $\mathcal{I}(\lceil \mathbf{Y}_0 \rfloor$ is star-convex and it contains a maximal convex open ball of the injectivity radius. We compute the injectivity radius next. The injectivity radius at a certain point is found by minimizing the distance from the point to the cut locus. Its minimum is achieved when exactly one of the principal angles is $\frac{\pi}{2}$ while all remaining $(p-1)$ are zero. Thus, the radius of injectivity at every point is the same and is the radius of injectivity r of the manifold $\mathcal{G}r_{\mathfrak{n},\mathfrak{p}}$. It is given by:

$$r = \frac{\pi}{2} \tag{4.33}$$

Similarly, the normalized radius of injectivity is given by:

$$r_n = \frac{1}{\frac{\pi}{2}\sqrt{p}} \cdot \frac{\pi}{2} = \frac{1}{\sqrt{p}} \tag{4.34}$$

For example, for $p = 2$ we have $r_n \approx 0.7071$, and for $p = 3$ we have $r_n \approx 0.5774$ with the maximal normalized distance being 1. In the next section we study barycentric combination of points that lie in a convex regular open ball. So both the radius of injectivity and the radius of largest regular ball are quite important. They tell us how big the open ball can be for the barycentric combination to exist and be unique. Recall the result of Theorem 3.6.1. It gives a sufficient condition for the uniqueness of the Karcher mean as a bound in terms of the sectional curvature. For this reason, we first present bounds on the sectional curvature of the Grassmann manifold $\mathcal{G}r_{\mathfrak{n},\mathfrak{p}}$.

Lemma 4.5.4. *The sectional curvature of the real Grassmann manifold $\mathcal{G}r_{n,p}(\mathbb{R})$ satisfies:*

$$0 \leqslant K_{\mathcal{G}r_{n,p}} \leqslant 1 \tag{4.35}$$

We refer the reader for this result to [109], [102]. We should note that Wong in his paper, [109], gives a bound on the sectional curvature to be $K_{\mathcal{G}r_{\mathrm{n,p}}} \leqslant 2$. However, the canonical metric from Equation (4.16) in this work does not have the normalizing coefficient of $\frac{1}{2}$. This should explain the discrepancy between Equation (4.35) and Wong's bound. Formula for computation of the sectional curvature as well as bounds on it are given in [102]. Now we compute the radius of the largest regular ball.

Theorem 4.5.5. *Every open normal ball on the Grassmann manifold is a regular ball.*

Proof. Based on Theorem 3.6.1, the sufficient condition for the ball to be regular is given from the solution of $K_{\mathcal{G}r_{\mathrm{n,p}}} \leqslant 1 < \left(\frac{1}{r} \cdot \frac{\pi}{2}\right)^2$. So we get a condition $r < \frac{\pi}{2}$. But recall that the radius of injectivity is $\frac{\pi}{2}$ from Equation (4.33). Thus, every normal open ball, i.e., open ball of the radius less than $\frac{\pi}{2}$, will be regular as well. □

In this section we studied the maximal radius of the normal regular open ball on the Grassmann manifold. In a way, these are "convex" balls that enjoy nice properties. For example, harmonic maps with values in a "convex" balls have a regularity and uniqueness of weak solutions. In particular, barycentric combinations exist and are unique. Also, one could consider probability distributions supported on such balls as well as other related concepts.

Now, we redefine the barycentric combination of two points.

4.6 Barycentric combination of two points redefined

We start by giving an alternative definition of a barycenter of two points via their minimal geodesic segment. Then we show that the definition coincides with the one given in Section 3.6.

Definition: Let μ_1, μ_2 be scalars summing to 1, and $\lceil \mathbf{Y}_1 \rfloor, \lceil \mathbf{Y}_2 \rfloor \in \mathcal{G}r_{\mathrm{n,p}}$ be two points within the interior of each other. This guarantees that they can be connected by the unique minimal geodesic segment γ defined by the boundary values $\gamma(\tau_1) = \lceil \mathbf{Y}_1 \rfloor$ and $\gamma(\tau_2) = \lceil \mathbf{Y}_2 \rfloor$, see Section 3.5. We define their *"barycentric combination,"* or simply the *"barycenter,"* to

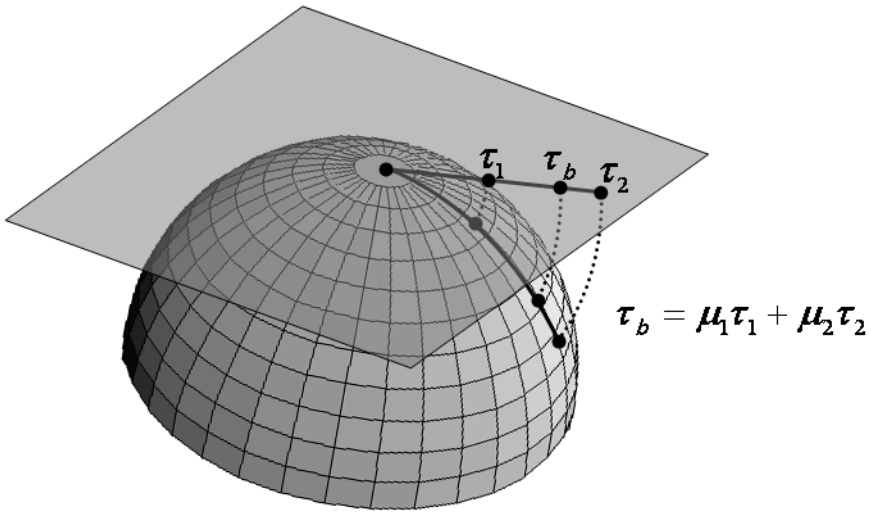

Fig. 4.3: Barycentric combination of two points

be the point $\lceil \mathbf{Y}_b \rfloor \triangleq \gamma(\mu_1\tau_1 + \mu_2\tau_2)$ and we write:

$$\lceil \mathbf{Y}_b \rfloor = \mu_1 \cdot \lceil \mathbf{Y}_1 \rfloor + \mu_2 \cdot \lceil \mathbf{Y}_2 \rfloor \tag{4.36}$$

We should note that when scalars μ_1, μ_2 are non-negative then both points $\lceil \mathbf{Y}_1 \rfloor$ and $\lceil \mathbf{Y}_2 \rfloor$ are guaranteed to be within the interior of the barycenter $\lceil \mathbf{Y}_b \rfloor$, so it will be well defined. However, when one of the scalars is taken largely negative, the geodesic segment γ could extend past the cut locus and will fail to be distance realizing. In such a case, the definition of the barycenter is ill defined. We establish the result in the following theorem.

Theorem 4.6.1 (sufficient condition). *The barycenter $\lceil \mathbf{Y}_b \rfloor$ defined in Equation (4.36) exists and coincides with the unique Karther mean if $\lceil \mathbf{Y}_1 \rfloor, \lceil \mathbf{Y}_2 \rfloor \in \mathcal{B}_r\big(\lceil \mathbf{Y}_b \rfloor\big)$ where r is the radius of the largest regular convex ball as given in Equation (4.33).*

Proof. First we need to check for the necessary condition given in Theorem 3.6.2. Essentially, we need to verify that all forces at $\lceil \mathbf{Y}_b \rfloor$ sum to zero. Let γ be the minimal geodesic segment connecting $\gamma(\tau_1) = \lceil \mathbf{Y}_1 \rfloor$ and $\gamma(\tau_2) = \lceil \mathbf{Y}_2 \rfloor$. Recall that the Grassmann manifold is complete. Thus, the geodesic γ gives rise to the global geodesic flow, per Section 3.3:

$$\gamma(t) = \gamma(\tau_1) \dot{+}_1 (t - \tau_1) \cdot \big(\gamma(\tau_2) \dot{-}_1 \gamma(\tau_1)\big)$$

On the other hand, every global flow is a one-parameter group action, see Theorem 3.2.2. Denote by $\boldsymbol{\Delta} = \big(\gamma(\tau_2) \ \dot{-}_{\gamma} \gamma(\tau_1)\big)$ and consider the following barycentric combination of forces:

$$\begin{aligned} F(\tau_b) &= \mu_1\Big(\gamma(\tau_1) \ \dot{-}_{\gamma} \gamma(\tau_b)\Big) + \mu_2\Big(\gamma(\tau_2) \ \dot{-}_{\gamma} \gamma(\tau_b)\Big) = \\ &= \mu_1\Big(- (\tau_b - \tau_1)\Big)\boldsymbol{\Delta} + \mu_2\Big((\tau_2 - \tau_1) - (\tau_b - \tau_1)\Big)\boldsymbol{\Delta} = \\ &= \Big(\mu_1\tau_1 - \mu_1\tau_b + \mu_2\tau_2 - \mu_2\tau_b\Big)\boldsymbol{\Delta} = 0 \cdot \boldsymbol{\Delta} = \mathbf{0}. \end{aligned}$$

On the other hand, since the ball $\mathcal{B}_r\big(\lceil \mathbf{Y}_b \rfloor\big)$ is convex and regular and contains both points, based on the sufficient condition in Theorem 3.6.1, the Karcher mean exists and is unique. So it must coincide with the critical point $\lceil \mathbf{Y}_b \rfloor$. □

We illustrate the definition of the barycentric combination of two points by Figure 4.3. In this figure, we show a geodesic $\gamma(t)$ connecting the two points $\gamma(\tau_1)$ and $\gamma(\tau_2)$. It is displayed as a curve on the half-sphere. We also show a tangent plane attached at a point $\gamma(0)$. An image of the geodesic on the tangent plane is a straight line. Then an affine combination of two points along this straight line will remain on this line. It will be a point reached at $\tau_b = \mu_1\tau_1 + \mu_2\tau_2$. It will correspond to the point $\gamma(\tau_b)$ on the manifold.

As one can see, the barycentric combination of two points requires obtaining a geodesic given initial and final values. This is a 2-point boundary value problem. We solve it next. We proceed as follows. First, we derive a linear and efficient solution for a barycentric combination of two points with equal weights. Then, we extend this solution to the barycentric combination of two points with arbitrary weights summing to one. In the later sections we also give an explicit linear algorithm for computing a barycentric combination of k points.

4.7 Linear solution for the midpoint of two points

Consider two points $\lceil \mathbf{Y}_0 \rfloor, \lceil \mathbf{Y}_\tau \rfloor \in \mathcal{G}r_{\mathrm{np}}$ on the Grassmann manifold that lie within the interior of each other with the distance $\tau = \mathrm{d}\Big(\lceil \mathbf{Y}_0 \rfloor, \lceil \mathbf{Y}_\tau \rfloor\Big)$ in between them. Let $\gamma(t)$ be the minimal geodesic segment connecting these two points with the initial and final values given by $\gamma(0) = \lceil \mathbf{Y}_0 \rfloor$ and $\gamma(\tau) = \lceil \mathbf{Y}_\tau \rfloor$. The following lemma gives equations for the mid-point $\lceil \mathbf{Y}_m \rfloor = \gamma(\frac{\tau}{2})$ of the minimal geodesic segment and its velocity vector $\lceil \mathbf{H}_m \rfloor = \gamma'(\frac{\tau}{2})$ at the mid-point.

Lemma 4.7.1. *The equation for the geodesic velocity vector field $\dot{\gamma}\big|_{\gamma}$ at time $\frac{\tau}{2}$ with the initial and final values at times $t = 0$ and $t = \tau$ is given by:*

$$\begin{array}{rcll} \gamma(\frac{\tau}{2}) = \lceil \mathbf{Y}_m \rfloor & = & \frac{1}{2}\Big(\lceil \mathbf{Y}_\tau \rfloor \cdot V_\tau + \lceil \mathbf{Y}_0 \rfloor \cdot V_0 \Big) \cdot C_m^{-1} V_0^T & \gamma(0) = \lceil \mathbf{Y}_0 \rfloor \\ \gamma'(\frac{\tau}{2}) = \lceil \mathbf{H}_m \rfloor & = & \frac{1}{2}\Big(\lceil \mathbf{Y}_\tau \rfloor \cdot V_\tau - \lceil \mathbf{Y}_0 \rfloor \cdot V_0 \Big) \cdot S_m^{-1} E V_0^T & \gamma(\tau) = \lceil \mathbf{Y}_\tau \rfloor \end{array} \tag{4.37}$$

where the rotation matrices V_0, V_τ are found from an SVD decomposition:

$$\mathbf{Y_0}^T \mathbf{Y}_\tau = V_0 C_\tau V_\tau^T \tag{4.38}$$

and (based on Section 4.5) C_τ is a cosine of principal angles. Define $E_\tau = \operatorname{acosm}(C_\tau)$, then compute the distance $\tau = \|E_\tau^2\|_F = \sqrt{\operatorname{trace}(E_\tau^2)}$ and a unit-distance matrix of principal angles $E \triangleq E_\tau / \tau$. In addition, define $C_m = \operatorname{cosm}(E\frac{\tau}{2})$ and $S_m = \operatorname{sinm}(E\frac{\tau}{2})$.

Proof. Recall, from the Equation (3.37) of the geodesic we have $\gamma(\tau) = \gamma(0) \cdot \exp\big(K\tau\big)$, where K is of the form given in Equation (4.7). Then we can write:

$$\left[\begin{array}{c|c} \mathbf{Y}_\tau & \mathbf{Y}_{\tau\perp} \end{array}\right] = \left[\begin{array}{c|c} \mathbf{Y_0} & \mathbf{Y_{0\perp}} \end{array}\right] \cdot \exp\left(\begin{bmatrix} F & -R^T \\ R & G \end{bmatrix} \tau \right) \tag{4.39}$$

But, as it was shown in Equation (4.12), vertical movements only result in two independent rotations in the subspace of $\mathbf{Y}_\tau$ and in the subspace of $\mathbf{Y}_{\tau\perp}$. Thus, we can write:

$$\begin{bmatrix} \mathbf{Y_0}^T \mathbf{Y}_\tau & \mathbf{Y_0}^T \mathbf{Y}_{\tau\perp} \\ \mathbf{Y_{0\perp}}^T \mathbf{Y}_\tau & \mathbf{Y_{0\perp}}^T \mathbf{Y}_{\tau\perp} \end{bmatrix} = \exp\left(\begin{bmatrix} 0 & -R^T \\ R & 0 \end{bmatrix} \tau \right) \cdot \begin{bmatrix} V_p & 0 \\ 0 & V_{n-p} \end{bmatrix} \tag{4.40}$$

where V_p and V_{n-p} are the two independent rotations in the subspace of $\mathbf{Y}_\tau$ and in the subspace of $\mathbf{Y}_{\tau\perp}$. Applying Equation (4.21) we obtain:

$$\left[\begin{array}{c|c} V_0^T \mathbf{Y_0}^T \mathbf{Y}_\tau V_\tau & \cdots \\ \hline \cdots & V_{0\perp}^T \mathbf{Y_{0\perp}}^T \mathbf{Y}_{\tau\perp} V_{\tau\perp} \end{array}\right] = \left[\begin{array}{c|cc} C_\tau & -S_\tau & 0 \\ \hline S_\tau & C_\tau & 0 \\ 0 & 0 & I \end{array}\right] \tag{4.41}$$

where V_0, V_τ, $V_{0\perp}$, and $V_{\tau\perp}$ are combined orthogonal rotations from Equations (4.40) and (4.21). This verifies the ability of writing an SVD decomposition given in Equation (4.38).

Now, we will derive equations for the mid-point $\lceil \mathbf{Y_m} \rfloor = \gamma(\frac{\tau}{2})$ of the geodesic and the tangent vector $\lceil \mathbf{H_m} \rfloor = \gamma'(\frac{\tau}{2})$ at mid-point. We take the equation of the geodesic in Equation (4.18) and write:

$$\lceil \mathbf{Y}_\tau \rfloor = \lceil \mathbf{Y_0} \rfloor \cdot V C_\tau V^T + \lceil \mathbf{H_0} \rfloor \cdot V S_\tau E^{-1} V^T \tag{4.42}$$

By taking into consideration the SVD decomposition given in Equation (4.38) we obtain:

$$\Big(\mathbf{Y}_\tau V_\tau \Big) = \Big(\mathbf{Y_0} V_0 \Big) \cdot C_\tau + \Big(\mathbf{H_0} V_0 E^{-1} \Big) \cdot S_\tau \tag{4.43}$$

Now we consider the rotated sum and difference of $\mathbf{Y}_\tau$ and $\mathbf{Y_0}$ as follows:

$$\begin{array}{rcl} \Big(\mathbf{Y}_\tau V_\tau\Big) + \Big(\mathbf{Y_0} V_0\Big) & = & \Big(\mathbf{Y_0} V_0\Big) \cdot \Big(I + C_\tau\Big) + \Big(\mathbf{H_0} V_0 E^{-1}\Big) \cdot S_\tau \\ \Big(\mathbf{Y}_\tau V_\tau\Big) - \Big(\mathbf{Y_0} V_0\Big) & = - & \Big(\mathbf{Y_0} V_0\Big) \cdot \Big(I - C_\tau\Big) + \Big(\mathbf{H_0} V_0 E^{-1}\Big) \cdot S_\tau \end{array} \tag{4.44}$$

We also have the trigonometric identities $2C_m^2 = I + C_\tau$, $2S_m^2 = I - C_\tau$, and $2C_m S_m = S_\tau$ in place. Then after substitution we can write:

$$\begin{array}{rcl} \frac{1}{2}\Big(\mathbf{Y}_\tau V_\tau + \mathbf{Y_0} V_0\Big) & = & \Big(\mathbf{Y_0} V_0\Big) \cdot C_m C_m + \Big(\mathbf{H_0} V_0 E^{-1}\Big) \cdot C_m S_m \\ \frac{1}{2}\Big(\mathbf{Y}_\tau V_\tau - \mathbf{Y_0} V_0\Big) & = - & \Big(\mathbf{Y_0} V_0\Big) \cdot S_m S_m + \Big(\mathbf{H_0} V_0 E^{-1}\Big) \cdot C_m S_m \end{array} \tag{4.45}$$

By noting that matrices C_m, S_m and E are diagonal and commute we wrire:

$$\begin{array}{rcl} \frac{1}{2}\Big(\mathbf{Y}_\tau V_\tau + \mathbf{Y_0} V_0\Big) C_m^{-1} V_0^T & = & \Big(\mathbf{Y_0} V_0\Big) \cdot C_m V_0^T + \Big(\mathbf{H_0} V_0 E^{-1}\Big) \cdot S_m V_0^T \\ \frac{1}{2}\Big(\mathbf{Y}_\tau V_\tau - \mathbf{Y_0} V_0\Big) S_m^{-1} E V_0^T & = - & \Big(\mathbf{Y_0} V_0\Big) \cdot S_m E V_0^T + \Big(\mathbf{H_0} V_0 E^{-1}\Big) \cdot C_m E V_0^T \end{array} \tag{4.46}$$

By inspection of the right hand side of the Equation (4.46) with the Equation (4.18) we see that the left hand side of the Equation (4.46) is the value of the geodesic at time $t = \frac{\tau}{2}$. This completes the proof. As a side note, we should say that the rotation by V_0 applied to right hand side of Equation (4.37) is optional. It can be any other rotation matrix. When applied, it makes the matrix $\mathbf{Y_m}^T \mathbf{Y_0}$ to be symmetric. For example, if we took V_τ instead, then the matrix $\mathbf{Y_m}^T \mathbf{Y}_\tau$ would be symmetric. Another example is to take a half-rotation matrix in between V_0 and V_τ for the purposes of charting a mid-point of two shapes. □

So far we have given a linear solution to the computation of the barycentric combination of two points with equal weights. Moreover, we have expressed the geodesic velocity vector field $\dot{\gamma}\big|_\gamma$ at the midpoint in terms of the initial and the final value of the minimal geodesic segment connecting them, as given in Equation (4.37). This expression gives rise to the solution of the logarithmic map, when used as an initial condition to the equation of the exponential map, given in Equation (4.18). We present these results next.

4.8 Linear solution for the geodesic segment — the logarithmic map

The problem we would like to solve is quite different from the solution given by Equation (4.18). Instead of defining an equation for the geodesic velocity vector field, given an initial point and a tangent direction at the point, it is essential for us to define these equations, given an initial point and a final point on the $\mathcal{G}r_{\mathrm{np}}$, which are to be connected by a geodesic. In the theory of differential equations this corresponds to a *two-point boundary*

value problem. Such a problem is commonly solved numerically, using the shooting method [86]. We present our result for an explicit expression for the solution of the two-point boundary value problem involving only standard linear algebra operations. Also recall from Section 3.5 that this is also a linear solution for the logarithmic map, which is known to be a very hard problem, [1]. This result first appeared in our publication in [98].

As before, consider two points $\lceil \mathbf{Y}_0 \rfloor, \lceil \mathbf{Y}_\tau \rfloor \in \mathcal{G}r_{\mathfrak{np}}$ on the Grassmann manifold that lie within the interior of each other with the distance $\tau = \mathrm{d}\Big(\lceil \mathbf{Y}_0 \rfloor, \lceil \mathbf{Y}_\tau \rfloor\Big)$ in between them. Let $\gamma(t)$ be the minimal geodesic segment connecting these two points with the initial and final values given by $\gamma(0) = \lceil \mathbf{Y}_0 \rfloor$ and $\gamma(\tau) = \lceil \mathbf{Y}_\tau \rfloor$. The following lemma gives an equation for the the velocity vector $\lceil \mathbf{H}_0 \rfloor = \gamma'(0)$ at the point $\lceil \mathbf{Y}_0 \rfloor$ that defines the geodesic segment when taken as an initial condition to Equation (4.18).

Lemma 4.8.1. *The equation for the geodesic velocity vector $\gamma'(0)$ for the geodesic segment $\dot{\gamma}\big|_\gamma$ defined by the initial and final values $\gamma(0)$ and $\gamma(\tau)$ is given by:*

$$\mathbf{H}_0 = -\mathbf{Y}_0 V_0 C_\tau {S_\tau}^{-1} E V_0^T + \mathbf{Y}_\tau V_\tau {S_\tau}^{-1} E V_0^T \tag{4.47}$$

where the rotation matrices V_0, V_τ are found from an SVD decomposition ${\mathbf{Y_0}}^T \mathbf{Y}_\tau = V_0 C_\tau V_\tau^T$, given in Equation (4.38), and $E = \mathrm{acosm}(C_\tau)/\tau$ with $S_\tau = \mathrm{sinm}(E{\cdot}\tau)$ and $C_\tau = \mathrm{cosm}(E{\cdot}\tau)$.

Proof. In order to obtain a closed form solution for the $\mathbf{H}_0$, all we need to do is to take an expression for $\mathbf{Y}_m$ and $\mathbf{H}_m$ from Equation (4.37) as an initial values $\gamma(0) = \lceil \mathbf{Y}_m \rfloor$ and $\gamma'(0) = \lceil \mathbf{H}_m \rfloor$ to the geodesic defined by Equation (4.18) and obtain the values for $\mathbf{H}_t$ at time $t = -\frac{\tau}{2}$. The following expression is obtained from Equation (4.37):

$$\begin{array}{rcl} \mathbf{Y}_m V_0 & = & \frac{1}{2}\Big(\mathbf{Y}_\tau \cdot V_\tau + \mathbf{Y}_0 \cdot V_0\Big) \cdot C_m^{-1} \\ \mathbf{H}_m V_0 E^{-1} & = & \frac{1}{2}\Big(\mathbf{Y}_\tau \cdot V_\tau - \mathbf{Y}_0 \cdot V_0\Big) \cdot S_m^{-1} \end{array} \tag{4.48}$$

Now we rewrite the second line of Equation (4.18) in the following form:

$$\mathbf{H}_t V_0 E^{-1} = -\mathbf{Y}_0 V_0 S_t + \mathbf{H}_0 V_0 C_t E^{-1} \tag{4.49}$$

Now we reverse time and evaluate this expression at $t = -\frac{\tau}{2}$. We get:

$$\mathbf{H}_0 V_0 E^{-1} = +\Big(\mathbf{Y}_m V_0\Big) S_m + \Big(\mathbf{H}_m V_0 E^{-1}\Big) C_m \tag{4.50}$$

Note that since time is negative, based on the symmetry properties of sine and cosine, we take S_m with minus and C_m with plus in the above expression. Now we plug in an expression

from Equation (4.48) into the expression above and obtain:

$$\mathbf{H}_0 V_0 E^{-1} = +\frac{1}{2}\Big(\mathbf{Y}_\tau V_\tau + \mathbf{Y}_0 V_0\Big) C_m^{-1} S_m + \frac{1}{2}\Big(\mathbf{Y}_\tau V_\tau - \mathbf{Y}_0 V_0\Big) S_m^{-1} C_m \tag{4.51}$$

Since the diagonal matrices commute, we are going to abuse the notation and write $\frac{S_m}{C_m} = C_m^{-1} S_m$. Then by regrouping we obtain:

$$\mathbf{H}_0 V_0 E^{-1} = \frac{1}{2}\mathbf{Y}_0 V_0 \Big(\frac{S_m}{C_m} - \frac{C_m}{S_m}\Big) + \frac{1}{2}\mathbf{Y}_\tau V_\tau \Big(\frac{S_m}{C_m} + \frac{C_m}{S_m}\Big) \tag{4.52}$$

By further simplifying we get:

$$\mathbf{H}_0 V_0 E^{-1} = \mathbf{Y}_0 V_0 \frac{S_m^2 - C_m^2}{2 C_m S_m^2} + \mathbf{Y}_\tau V_\tau \frac{S_m^2 + C_m^2}{2 C_m S_m^2} \tag{4.53}$$

Using the trigonometric identities we finally get:

$$\mathbf{H}_0 V_0 E^{-1} = -\mathbf{Y}_0 V_0 \frac{C_\tau}{S_\tau} + \mathbf{Y}_\tau V_\tau \frac{1}{S_\tau} \tag{4.54}$$

This gives the desired result. □

We note that $\mathbf{H}_0$ obtained in Equation (4.47) will be of the unit norm, i.e., $\|\mathbf{H}_0\|_F = 1$. We also present another alternative way for solving for $\mathbf{H}_0$. If we use brackets to denote stacking of matrices together to form a larger matrix, then an alternative solution can be obtained from the equation:

$$\Big[\ \mathbf{Y}_0 \Big|\ \mathbf{Y}_\tau\ \Big]^T \cdot \Big[\ \mathbf{H}_0 \Big|\ \mathbf{H}_\tau\ \Big] = \begin{bmatrix} V_0 & 0 \\ 0 & V_\tau \end{bmatrix} \cdot \begin{bmatrix} 0 & -ES_\tau \\ ES_\tau & 0 \end{bmatrix} \cdot \begin{bmatrix} V_0 & 0 \\ 0 & V_\tau \end{bmatrix}^T \tag{4.55}$$

From this equation we simultaneously solve for $\mathbf{H}_0$ and $\mathbf{H}_\tau$ using the pseudo-inverse of $[\mathbf{Y}_0|\mathbf{Y}_\tau]^T$. These two tangents are unit norm parallel-transported versions of each other, and belong to the geodesic's derivative vector field.

We now present our result for the equations of the geodesic $\gamma(t) = \lceil \mathbf{Y}_t \rfloor$ and its derivative vector field along the geodesic $\gamma'(t) = \lceil \mathbf{H}_t \rfloor$ at time t. This is an explicit expression for the solution of the two-point boundary value problem.

Theorem 4.8.2. *The equation for the geodesic velocity vector field $\dot{\gamma}\big|_\gamma$ for the minimal geodesic segment $\dot{\gamma}\big|_\gamma$ defined by the initial and final values $\gamma(0) = \lceil \mathbf{Y}_0 \rfloor$ and $\gamma(\tau) = \lceil \mathbf{Y}_\tau \rfloor$ is given by:*

$$\begin{aligned} \mathbf{Y}_t =&\ \mathbf{Y}_0 V_0 (C_t - S_t C_\tau S_\tau{}^{-1}) V_0^T + \mathbf{Y}_\tau V_\tau S_t S_\tau^{-1} V_0^T \\ \mathbf{H}_t =& -\mathbf{Y}_0 V_0 (S_t + C_t C_\tau S_\tau^{-1}) E V_0^T + \mathbf{Y}_\tau V_\tau C_t S_\tau^{-1} E V_0^T \end{aligned} \tag{4.56}$$

This is an explicit solution for the two-point boundary value problem. The time-varying matrices in equation (4.56) are $C_t = cosm(t{\cdot}E)$ and $S_t = sinm(t{\cdot}E)$. All other matrices are obtained from an SVD decomposition of the inner product $\mathbf{Y}_0^T\mathbf{Y}_\tau$, given in Equation (4.38).

Proof. In order to prove the theorem and obtain a closed form solution for the geodesic and its derivative vector field for the two-point boundary value problem, we take an expression of $\mathbf{H}_0$ obtained in Equation (4.47) and substitute it into the Equation (4.18). After simplifications we get the desired result. □

Equation (4.56) gives a general form for the logarithmic map. Its computation is very efficient and robust. Essentially, we are computing an SVD decomposition of the inner product $\mathbf{Y}_0^T\mathbf{Y}_\tau$, which is an p-by-p matrix with a typical values of $p = 2, 3$. Then we are computing some sines and cosines for diagonal p-by-p matrix E of principal angles. This is equivalent to computing sines and cosines of the elements along the diagonal. Finally we combine the matrices according to the Equation (4.56). The free parameter is the value of t. For example, by taking $t = 0$ we obtain the initial point $\lceil\mathbf{Y}_0\rfloor$, or by taking $t = \tau$ we obtain the final point $\lceil\mathbf{Y}_\tau\rfloor$. The midpoint is obtained by taking $t = \frac{\tau}{2}$. Any barycentric combination of the initial and the final points is obtained by varying t according to Equation (4.36).

Now we present an algorithm for computing barycentric combinations of more than two points. It relies on a solution for a barycentric combination of two points recursively by induction.

4.9 Linear algorithm for the barycentric combination of k points

Equation (4.56) and Equation (4.36) tell us how to efficiently compute a barycentric combination of two points. The computation is linear and explicit. It turns out that an approximation to the barycentric combination of $k > 2$ points is simply obtained by recursively dividing points into smaller subgroups, computing the barycentric combination of each of the groups, and then computing barycentric combinations of the groups with appropriately chosen weights. We establish the result in the following conjecture:

Conjecture 4.9.1 (barycentric combination of k points). *Let $\left(\lceil \mathbf{Y}_i \rfloor\right)_{i\in I}$ be a constellation of points in $\mathcal{G}r_{n,p}$, and let $(\mu_i)_{i\in I}$ be a family of scalars summing to 1. Suppose the barycenter $\lceil \mathbf{Y}_b \rfloor = \sum_{i\in I} \mu_i \lceil \mathbf{Y}_i \rfloor$ exists and is unique. Recall that a sufficient condition for this is given in Threoem 3.6.1. Then the + operator is approximately associative. In other words, if we split the set I into two (or more) arbitrary disjoint subsets $I_1 \cup I_2 = I$ and $I_1 \cap I_2 = \emptyset$ and denote the new weights by $\Omega_1 = \sum_{i\in I_1} \mu_i$ and $\Omega_2 = \sum_{i\in I_2} \mu_i$, then:*

$$\lceil \mathbf{Y}_b \rfloor \approx \Omega_1 \left(\sum_{i\in I_1} \frac{\mu_i}{\Omega_1} \lceil \mathbf{Y}_i \rfloor \right) + \Omega_2 \left(\sum_{i\in I_2} \frac{\mu_i}{\Omega_2} \lceil \mathbf{Y}_i \rfloor \right) \tag{4.57}$$

The conjecture has been verified experimentally. One can choose an arbitrary parenthesization and recursively compute the barycentric combinations till each group is left with two points. Then compute barycentric combination of two points using the logarithmic map, as given by Equation (4.56) and by Equation (4.36). Finally, one verifies the necessary condition given in Theorem 3.6.2. If one attaches the tangent space at the computed barycenter, then the forces given in Equation (3.27) must sum to 0. Based on our experiments, the integral force will always be very close to 0. Thus, this algorithm gives an efficient linear approximation to the computation of barycentric computations. Recall from Section 3.6 that, in order to achieve exact results, one could also use a non-linear minimization of the cost function, given in Equation (3.26). However, depending on the application, our linear approximation might provide sufficiently good results.

Computation of the barycentric combinations entirely relies on the logarithmic map — a solution to the two point boundary value problem. It shows the significance of being able to efficiently calculate the logarithmic map. The both results for efficiently and analytically calculating the logarithmic map and the barycentric combinations are new for the Grassmann manifold and are not known in the literature. Typical approaches taken by others is to use some sort of non-linear optimization in order to find a solution to the logarithmic map and to the barycentric combinations, see for example [88].

4.10 Sample mean and higher order statistics

Theorem 4.9.1 gives a linear algorithm for computing the barycentric combinations of k points. In addition, we know that this computation will succeed as long as all points lie within the radius of injectivity from the computed barycenter. Recall that the radius of injectivity is given in Equation (4.33). In particular, if we take equal weights $\mu_i = \frac{1}{k}$, then we obtain a computation of the sample mean:

$$\lceil \mathbf{Y}_m \rfloor = \sum_{i=1}^{k} \frac{1}{k} \lceil \mathbf{Y}_i \rfloor \tag{4.58}$$

By attaching a tangent plane at $\lceil \mathbf{Y}_m \rfloor$ we can obtain a diffeomorphic representation for each of the points $\lceil \mathbf{Y}_i \rfloor$ by the tangent vectors $\left(\lceil \mathbf{Y}_i \rfloor \multimap \lceil \mathbf{Y}_m \rfloor \right)$, given via the logarithmic map. The tangent space is a vector space, and we can compute sample mean and higher order statistics on it. In particular, one easily verifies that based on Equation (3.27) the sample mean of tangent vectors $\left(\lceil \mathbf{Y}_i \rfloor \multimap \lceil \mathbf{Y}_m \rfloor \right)$ is zero. Now, one can compute other higher order central moments that will already have the mean implicitly removed. Thus, one can do any kind of statistical computations on the tangent space attached at the sample mean. As an example, we can compute the covariance matrix, and then use it to define the Mahalanobis distance from the mean to the sample. For more examples, see [102].

4.11 Summary

In this chapter we introduced the numerical representation of points on the real Grassmann manifold $\mathcal{G}r_{\mathrm{np}}$. We gave the decomposition of the Grassmann manifold as a quotient of the orthogonal group and explicit analytical linear solutions for the exponential map and the logarithmic map. Closed form linear solutions for the midpoint of two points, for the geodesic segment and its derivative vector field were obtained. Also, we developed a linear and efficient algorithm for the computation of the approximate barycentric combinations of points on the Grassmann manifold and discussed related to it notions of principal angles, cut locus, and maximal radii for normal regular balls. Our efficient solutions for the logarithmic map and the approximate barycentric combinations are new for the Grassmann manifold

and are not known in the literature.

The main contribution of the chapter is the development of original formulas for computations of such important characteristics as distance between two points, the logarithmic map as a two point boundary value problem, sample mean of k points, etc. The obtained results are directly applied to the affine-invariant shape space, allowing computation of similarity in between the shapes. It can be noted that the results can be used in many other theories besides the shape representation theory.

5. $\mathcal{G}^{\mathcal{AP}}_{\mathbf{n,p}}$-INVARIANT SHAPE SPACE AS LOCALLY GRASSMANN MANIFOLD

5.1 Preliminaries

This chapter addresses the problems of representation for the affine-permutation invariant shape space and computations on it. Our goal in this research is to find a geometric manifold $\mathcal{GS}_{\mathrm{n,p}}$ with an unlabeled-view-independent metric properly defined on it. This is a very challenging problem.

The key points that will be addressed in this chapter are:

- ***Non-existence of global geometry in $\mathcal{GS}_{n,p}$— Section 5.2***
- ***Invariance to permutations as an optimization problem — Sections 5.3-5.4***
- ***Survey of alternative solutions to permutations — Section 5.5***
- ***Analyzing the geometric cost function — Section 5.6***
- ***An approximation to optimization via QP and LP — Sections 5.7-5.8***
- ***Canonical representation of a set and local geometry of $\mathcal{GS}_{n,p}$— Section 5.9***

In this chapter we start by discussing as to why the quotient space $\mathcal{GS}_{\mathrm{n,p}} = \mathcal{G}r_{\mathrm{n,p}}/\mathcal{P}_{\mathrm{n}}$ is not a Riemannian manifold. Then, in order to achieve an invariance to permutations, we look for a similarity measure in the residual shape space $\mathcal{GS}_{\mathrm{n,p}}$ as an optimization problem. This similarity measure should be invariant to both affine transformations and permutations. It should also be an easily computable metric. We formulate the combinatorial optimization over the finite group of permutations as a relaxed convex hull optimization over the doubly stochastic matrices. We present the reader with a survey of alternative ways of dealing with permutation distortions that are frequently used in the literature. Then we go back to our

optimization problem and discuss why optimization of the true geometric distance between two points is not easily computable. We derive two algorithms for the computation of an approximation to the distance. One is based on quadratic programming, and the second one is based on linear programming.

Based on our experiments, we establish a conjecture. It states that if the minimized distance found as a result of linear programming is large, then the true minimized distance would have been large as well. In other words, it tells us that the linear programming solution is a mapping that preserves distances locally. For this reason, given two shapes, we can establish how similar the shapes are, and if they are sufficiently similar, we can compute their distance.

In addition, in order to address the issue of defining a geometry in the shape space $\mathcal{GS}_{\mathrm{np}}$, we state another conjecture: if a constellation of shapes is sufficiently similar to each other, then there exists a diffeomorphic mapping of the constellation of shapes to a neighborhood of the Grassmann manifold. This makes the shape space $\mathcal{GS}_{\mathrm{np}}$ to locally look like the Grassmann manifold. Thus, we will have all the constructs from Chapter 4 applicable to sufficiently similar shapes. In particular, distance and the barycenter of shapes can be computed using the results in that chapter.

5.2 Non-existence of global geometry in $\mathcal{GS}_{np}$

As we discussed in Section 2.9, Figure 2.2 represents the mapping from a set of abstract shapes, given in block one of Figure 2.2, to a geometric space that achieves invariance to both affine transformations and permutations, given in block eight of Figure 2.2. According to this figure, the seventh block is the Grassmann manifold $\mathcal{G}r_{\mathrm{np}}$. This is a shape space that achieves invariance to affine tranformations only. This shape space possesses a global geometry — a geometry of the Grassmann manifold. Its geometry has been studied in Chapter 4.

The transition from the seventh block of Figure 2.2 to the eighth block is removing the ambiguity $\mathcal{P}_{\mathrm{n}}$ of arbitrary permutations. We allow arbitrary permutations of n feature

points, which are row permutations of the configuration matrices. In other words, if $\lceil \mathbf{Y} \rfloor \in \mathcal{G}r_{\mathfrak{n}\mathfrak{p}}$ is a point on the Grassmann manifold, we are allowing arbitrary row permutations of the matrix $\mathbf{Y}$. This ambiguity is eliminated by taking a finite group of permutations $\mathcal{P}_{\mathrm{n}}$ and folding the Grassmann manifold into $\mathcal{GS}_{\mathfrak{n}\mathfrak{p}} = \mathcal{G}r_{\mathfrak{n}\mathfrak{p}}/\mathcal{P}_{\mathrm{n}}$. Once this ambiguity has been removed, $\mathcal{GS}_{\mathfrak{n}\mathfrak{p}}$ is a shape space that achieves invariance to both affine tranformations and permutations.

The shape space $\mathcal{GS}_{\mathfrak{n}\mathfrak{p}} = \mathcal{G}r_{\mathfrak{n}\mathfrak{p}}/\mathcal{P}_{\mathrm{n}}$ cannot be represented as an embedded sub-manifold of the Riemannian manifold $\mathcal{G}r_{\mathfrak{n}\mathfrak{p}}$. We show next why this is the case.

Recall from Section 2.4 the definition of group action. In a way, a group action on a space is merely a rule, compatible with the group structure, by which group elements move points in the space. Now we define a free group action.

Definition: A group action is called *"free action"* if only the identity element of the group can leave a point fixed.

Given a group action on a space, we can form a quotient space by identifying and gluing together each point and its images under all of the group elements. Only if the group action of a finite group is free, the quotient space will again be a manifold, [16].

The finite group $\mathcal{P}_{\mathrm{n}}$ of permutations does not act freely on $\mathcal{G}r_{\mathfrak{n}\mathfrak{p}}$. To see this, take a matrix $\mathbf{Y} \in \mathcal{G}f_{\mathfrak{n}\mathfrak{p}}$ with first two rows being equal, and apply a permutation that swaps the first two rows. This permutation is a non-identity element of $\mathcal{P}_{\mathrm{n}}$, but it leaves $\mathbf{Y}$ unchanged. Thus, $\mathcal{P}_{\mathrm{n}}$ does not act freely on a compact Stiefel manifold $\mathcal{G}f_{\mathfrak{n}\mathfrak{p}}$. As a consequence, it does not act freely on the Grassmann manifold $\mathcal{G}r_{\mathfrak{n}\mathfrak{p}} = \mathcal{G}f_{\mathfrak{n}\mathfrak{p}}/\mathcal{O}_{\mathrm{p}}$ either.

We conclude that the quotient space $\mathcal{GS}_{\mathfrak{n}\mathfrak{p}} = \mathcal{G}r_{\mathfrak{n}\mathfrak{p}}/\mathcal{P}_{\mathrm{n}}$ is not an embedded Riemannian sub-manifold of $\mathcal{G}r_{\mathfrak{n}\mathfrak{p}}$. It does not possess the global Riemannian geometry intrinsically derived from $\mathcal{G}r_{\mathfrak{n}\mathfrak{p}}$ via the embedding.

Even though $\mathcal{GS}_{\mathfrak{n}\mathfrak{p}}$ does not possess a global geometry as a quotient geometry of $\mathcal{G}r_{\mathfrak{n}\mathfrak{p}}/\mathcal{P}_{\mathrm{n}}$, later in this chapter we will give a conjecture that will define a local geometry in the neighborhoods of points, allowing to perform computations of similarity as well as other geometric concepts when points are sufficiently close to each other.

Now we present invariance to permutations as an optimization problem.

5.3 Invariance to permutations as an optimization problem

Recall from Section 2.9 that whenever we consider factor spaces we form equivalency classes. Then in order to factor out an action of the group under consideration, we either obtain a canonical member for each of the equivalency classes, or stay within the space of orbits and derive formulas that are invariant to the choice of the representative of each equivalency class. As we showed in Section 5.2, the shape space $\mathcal{GS}_{\mathrm{n,p}} = \mathcal{G}r_{\mathrm{n,p}}/\mathcal{P}_{\mathrm{n}}$ does not admit a global geometry derived from $\mathcal{G}r_{\mathrm{n,p}}$. For this reason, we solve for the permutation invariance by means of finding a canonical member.

We start by defining similarity between two shapes in the shape space $\mathcal{GS}_{\mathrm{n,p}}$ as the solution to the distance minimization problem, where the distance is the Grassmannian distance from Equation (4.30), and by minimizing it over the group of permutations $\mathcal{P}_{\mathrm{n}}$. Then we give an algorithm for labeling points on the Grassmann manifold with permutation matrices as their labels. Finally, we find canonical members by applying the inverse permutation to the one stored in the label of each point. We proceed as follows.

Let $\lceil \mathbf{Y} \rfloor \in \mathcal{G}r_{\mathrm{n,p}}$ be a point on the Grassmann manifold. Alternatively, based on Section 2.4 and the shorthand notation introduced in Section 4.2, we can write point $\lceil \mathbf{Y} \rfloor = \lceil \mathbf{Y} \rfloor_{\mathcal{O}_{\mathrm{p}}}$ as the left coset $\mathbf{Y}\mathcal{O}_{\mathrm{p}}$ of group $\mathcal{O}_{\mathrm{p}}$ with respect to the point $\mathbf{Y} \in \mathcal{G}f_{\mathrm{n,p}}$. Similarly, a point in the shape space will be represented by the right coset:

$$\mathcal{P}_{\mathrm{n}}\mathbf{Y}\mathcal{O}_{\mathrm{p}} \in \mathcal{GS}_{\mathrm{n,p}} \tag{5.1}$$

of group $\mathcal{P}_{\mathrm{n}}$ with respect to $\mathbf{Y}\mathcal{O}_{\mathrm{p}}$. In this chapter we will represent points in the shape space mostly in the form of Equation (5.1) as we have two group actions in the picture. This representation is also more intuitive since each permutation matrix is $n \times n$ and is being applied to the orthonormal $n \times p$ matrix $\mathbf{Y}$ by multiplication from the left in order to permute rows of $\mathbf{Y}$, while each matrix from $\mathcal{O}_{\mathrm{p}}$ is $p \times p$ and is being applied to $\mathbf{Y}$ by

multiplication from the right. Sometimes we will also use the following notation:

$$\mathcal{P}_{\mathrm{n}}\lceil \mathbf{Y} \rfloor_{\mathcal{O}_{\mathrm{p}}} \triangleq \mathcal{P}_{\mathrm{n}}\mathbf{Y}\mathcal{O}_{\mathrm{p}} \in \mathcal{GS}_{\mathrm{n,p}} \tag{5.2}$$

This notation tells that there are two groups acting on $\mathcal{G}f_{\mathrm{n,p}}$, where one group is acting from the left, and the other one is acting from the right. Now we are ready to proceed with the definition of the similarity between two shapes in the shape space $\mathcal{GS}_{\mathrm{n,p}}$.

Definition: Let $\mathcal{P}_{\mathrm{n}}\lceil \mathbf{Y}_0 \rfloor_{\mathcal{O}_{\mathrm{p}}}$ and $\mathcal{P}_{\mathrm{n}}\lceil \mathbf{Y}_\tau \rfloor_{\mathcal{O}_{\mathrm{p}}}$ be two points in $\mathcal{GS}_{\mathrm{n,p}}$. We define the distance between them to be given by:

$$\mathrm{d}_{\mathcal{GS}_{\mathrm{n,p}}}\Big(\mathcal{P}_{\mathrm{n}}\lceil \mathbf{Y}_0 \rfloor_{\mathcal{O}_{\mathrm{p}}}, \mathcal{P}_{\mathrm{n}}\lceil \mathbf{Y}_\tau \rfloor_{\mathcal{O}_{\mathrm{p}}}\Big) \triangleq \min_{P_0, P_\tau \in \mathcal{P}_{\mathrm{n}}} \mathrm{d}_{\mathcal{G}r_{\mathrm{n,p}}}\Big(P_0 \cdot \mathbf{Y}_0\mathcal{O}_{\mathrm{p}}, P_\tau \cdot \mathbf{Y}_\tau\mathcal{O}_{\mathrm{p}}\Big) \tag{5.3}$$

where we are minimizing over two permutation matrices $P_0, P_\tau \in \mathcal{P}_{\mathrm{n}}$.

It turns out that it is not necessary to simultaneously minimize over both permutation matrices. One of them can be fixed to be the identity permutation. The result is stated below.

Lemma 5.3.1. *The minimization in Equation (5.3) is equivalent to the following one:*

$$d_{\mathcal{GS}_{n,p}}\Big(\mathcal{P}_n\lceil \mathbf{Y}_0 \rfloor_{\mathcal{O}_p}, \mathcal{P}_n\lceil \mathbf{Y}_\tau \rfloor_{\mathcal{O}_p}\Big) = \min_{P \in \mathcal{P}_n} d_{\mathcal{G}r_{n,p}}\Big(\mathbf{Y}_0\mathcal{O}_p, P \cdot \mathbf{Y}_\tau\mathcal{O}_p\Big) \tag{5.4}$$

where we are minimizing over a finite group of permutation matrices $P \in \mathcal{P}_n$.

Proof. In order to observe this note the following identity:

$$\mathrm{d}_{\mathcal{G}r_{\mathrm{n,p}}}\Big(P \cdot \mathbf{Y}_0\mathcal{O}_{\mathrm{p}}, P \cdot \mathbf{Y}_\tau\mathcal{O}_{\mathrm{p}}\Big) = \mathrm{d}_{\mathcal{G}r_{\mathrm{n,p}}}\Big(\mathbf{Y}_0\mathcal{O}_{\mathrm{p}}, \mathbf{Y}_\tau\mathcal{O}_{\mathrm{p}}\Big) \tag{5.5}$$

This identity comes from the fact that the inner product $\mathbf{Y}_0^T P^T P \mathbf{Y}_\tau$ is identical with $\mathbf{Y}_0^T \mathbf{Y}_\tau$, and so the Grassmannian distance in Equation (4.30) does not change. Then one easily obtains:

$$\mathrm{d}_{\mathcal{G}r_{\mathrm{n,p}}}\Big(P_0 \cdot \mathbf{Y}_0\mathcal{O}_{\mathrm{p}}, P_\tau \cdot \mathbf{Y}_\tau\mathcal{O}_{\mathrm{p}}\Big) = \mathrm{d}_{\mathcal{G}r_{\mathrm{n,p}}}\Big(\mathbf{Y}_0\mathcal{O}_{\mathrm{p}}, P_0^{-1}P_\tau \cdot \mathbf{Y}_\tau\mathcal{O}_{\mathrm{p}}\Big) \tag{5.6}$$

In other words, both permutations can be folded into just one permutation. This completes the proof. □

So far we have defined distance between shapes to be the minimization problem given in Equation (5.4). Now we are going to give an example as to why minimizing the Grassmannian distance over a finite group permutations is the right approach for defining similarity between shapes in the shape space $\mathcal{GS}_{\mathrm{n,p}}$.

5.4 Why define similarity in $\mathcal{GS}_{n,p}$ as an optimization over $\mathcal{P}_n$?

We present one example here that shows why it is important to achieve invariance to permutations by minimizing the Grassmannian distance in Equation (4.30) over a finite group of permutations, as defined in Equation (5.4).

Consider an example given in Figure 5.1. We take two different shapes and connect them by a geodesic on the Grassmann manifold, as defined in Equation (4.56). Our initial shape $\lceil \mathbf{Y}_0 \rfloor$ is a filled triangle, and our final shape $\lceil \mathbf{Y}_\tau \rfloor$ is a filled circle. The normalized Grassmannian distance, per Equation (4.32), between them is:

$$\mathrm{d}_{\mathcal{G}r_{\mathrm{n,p}}}\Big(\lceil \mathbf{Y}_0 \rfloor, \lceil \mathbf{Y}_\tau \rfloor\Big) = 0.535 \tag{5.7}$$

Now we perform the optimization given in Equation (5.4) and find a permutation P that minimizes it. In Figure 5.2 we present the geodesic between the same initial shape $\lceil \mathbf{Y}_0 \rfloor$ and the permuted final shape $\lceil P{\cdot}\mathbf{Y}_\tau \rfloor$. The normalized Grassmannian distance between them is:

$$\mathrm{d}_{\mathcal{G}r_{\mathrm{n,p}}}\Big(\lceil \mathbf{Y}_0 \rfloor, \lceil P{\cdot}\mathbf{Y}_\tau \rfloor\Big) = 0.166 \tag{5.8}$$

which is now much smaller than for the shapes in Figure 5.1. If we visually compare the final shapes in Figure 5.1 and 5.2, we see no difference. This is because they only differ by a permutation in the order of their feature points. However, the intermediate shapes on these geodesics are very different. The ones in Figure 5.1 are more distorted. They look as if feature points are getting rotated on a spiral in order to bring features of the initial and final shapes into correspondence with each other.

This example shows that minimizing the Grassmannian distance over the group of permutations is the proper way of defining a similarity measure between the shapes.

In the following sections we look for a solution to the minimization problem we defined in Equation (5.4).

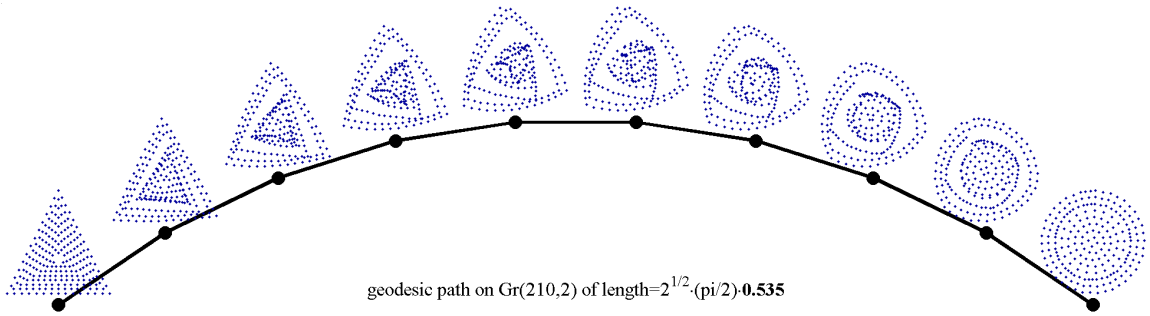

Fig. 5.1: Morphing a filled triangle onto a filled circle

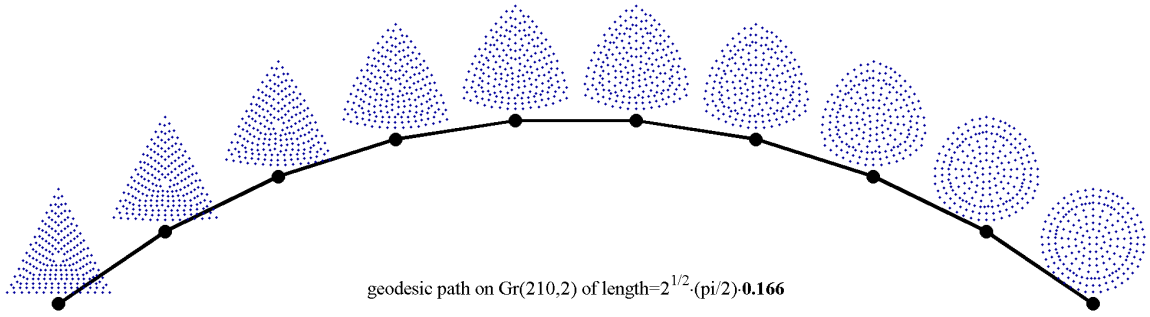

Fig. 5.2: Morphing a filled triangle onto a filled permuted circle

5.5 Survey of alternative solutions to the correspondence problem

There are alternative ways of defining distance between shapes besides the one defined by the minimization problem in Equation (5.4). In this section we give an overview of alternative approaches frequently considered in computer vision and signal processing communities.

Recall that permutation distortions arise from the fact that our feature points are not labeled, and thus they first need to be brought in correspondence with each other before the similarity of shapes can be evaluated. Bringing them in correspondence can be thought of as a combinatorial problem if one were to try each one of all possible permutations to find the best one. Recall that there are a total of $n!$ permutations for n features. Thus, this brute-force approach is not feasible for practical applications.

One way of dealing with permutations is by making feature points labeled. This is done in the following approach. Consider an example of a video sequence of an object. This is a set of views (images) densely spaced in time. Let us assume that the feature points have been found in the first view. Then it is reasonable to assume that the corresponding feature points in the following views are located at nearby locations. One can search locally for the displacement of these feature points. This type of algorithms are called feature tracking algorithms. As an example, see [99].

Another example to consider is to take a pair of stereo images. One could construct an $n \times n$ matrix of proximities and then look for correspondences in between the features using the Gestalt law of proximity. Such an algorithm is described in [97], where matching pairs are found by means of an SVD decomposition of the proximity matrix.

Alternatively, if the feature points can be represented by a column vector, then the problem of finding an optimal permutation is equivalent to the problem of sorting. Essentially, in order to bring two column vectors in correspondence with each other, both arrays need to be sorted in ascending order of their values, without loss of generality. Similarly, for $n \times p$ matrices one could define a canonical sorting to be sorting by the first column, with sub-sorting of the following columns in order to resolve ambiguities when values in the previous column coincide.

Yet another approach has been pioneered by Brockett, where combinatorial optimization is performed in differentiable setting, [18, 20, 19]. This approach, replaces the combinatorial search involved in finding the best permutation with a calculus problem involving orthogonal matrices. Essentially, an optimization over a group of permutations $\mathcal{P}_{\mathrm{n}}$ is relaxed to an optimization over an orthogonal group $\mathcal{O}_{\mathrm{n}}$, using the double-bracket continuous-time gradient flows. Additional reading can be found in [80, 24, 104, 57].

Finally, combinatorial optimization can also be performed as a constrained optimization over a convex hull. Under this framework, an optimization over a group of permutations $\mathcal{P}_{\mathrm{n}}$ is relaxed to an optimization over doubly stochastic matrices, which form a convex hull over the group of permutations, [73, 72]. Our approach follows this type of optimization.

Other theoretical approaches are to convert feature points into a space where permutation ambiguities disappear. As an example, consider the following idea. We can achieve an invariance to permutations by going from the roots of a polynomial to the polynomial itself. It is a well known fact that an unordered set of n roots uniquely defines the n-th order polynomial, thus achieving invariance to permutations. So if we take feature points to be roots, then a space of polynomials is a permutation-invariant representation for shape views. Related to the approach is to consider a Vandermonde matrix, and then compute its projectors, in order to achieve the desired invariance. Yet another idea in the same direction is to do the following. Suppose we have some open interval $\mathcal{I} \subset \mathbb{R}$, e.g. $\mathcal{I} = (-1; 1)$. Shapes can be parameterized by smooth atlases $\mathrm{s} : \mathcal{I} \to \mathbf{S} \subset \mathbb{R}^p$ such that $\mathrm{s}(I) = \mathbf{S}$ is the shape. We will refer to them as *"configuration atlases."* A set of all configuration atlases is a vector space. Assuming shapes have been already centered, call two atlases s_1 and s_2 equivalent if $\exists A \in \mathcal{G}L_{\mathrm{p}}$ such that $\mathrm{s}_1(\mathcal{I}) = A \cdot \mathrm{s}_2(\mathcal{I})$. Then this space forms a manifold, which is invariant to permutations when sampled. Sampling should take into the account distance along the curve between the sample points. Uniform sampling, in particular, produces the configuration matrices. For case $p = 2$, sampling points can be thought of as points in the complex plane. Then such sampling is possible due to the sampling theorem of complex functions.The approaches in this direction are primarily of theoretical importance than of

practical. They suffer from sensitivity to noise, and are hard to implement in practice.

Now we analyze the geometric cost function for the minimization problem in Equation (5.4).

5.6 Analyzing the geometric cost function

Let $\lceil \mathbf{Y}_0 \rfloor, \lceil P \cdot \mathbf{Y}_\tau \rfloor \in \mathcal{G}r_{\mathfrak{n}\mathfrak{p}}$ be two points on the Grassmann manifold. Then, recall from Section 4.5 that in order to compute the distance, we first perform the SVD decomposition:

$$\mathbf{Y}_0^T P \mathbf{Y}_\tau = V_0 C_\tau V_\tau^T \tag{5.9}$$

which gives two rotation matrices V_0, V_τ and a cosine diagonal matrix of the principal angles C_τ. Then we compute the matrix arc-cosine $E_\tau = \operatorname{acosm}(C_\tau)$. According to Equation (4.30), the distance is given by:

$$\mathrm{d}_{\mathcal{G}r_{\mathfrak{n}\mathfrak{p}}}\Big(\lceil \mathbf{Y}_0 \rfloor, \lceil P \cdot \mathbf{Y}_\tau \rfloor\Big) = \sqrt{\operatorname{trace}(E_\tau^2)} \tag{5.10}$$

Matrix sine, cosine, arc-sine, and arc-cosine are well defined for diagonal and symmetric matrices. We are going to extend the definition of these trigonometric functions to any arbitrary matrix A with the help of an SVD decomposition of A. This will help us to study the geometric distance on $\mathcal{G}r_{\mathfrak{n}\mathfrak{p}}$.

Definition: Let E be a diagonal matrix. We define the matrix exponential by:

$$e^E = \exp(E) \triangleq 1 + E/1 + E^2/2! + ... \tag{5.11}$$

Remark: Let Q be an orthonormal matrix. Then the matrix exponential satisfies:

$$\exp(QEQ^T) = Q \cdot \exp(E) \cdot Q^T \tag{5.12}$$

Definition: The matrix cosine and matrix sine of a diagonal matrix are defined by

$$\operatorname{cosm}(E) = \frac{e^{iE} + e^{-iE}}{2} \tag{5.13}$$

$$\operatorname{sinm}(E) = \frac{e^{iE} - e^{-iE}}{2i} \tag{5.14}$$

The matrix cosine and matrix sine of a diagonal matrix can be computed by applying scalar sine and cosine to the diagonal elements, while keeping off-diagonal elements at zero. The definition is naturally extended to symmetric matrices using the property above, given in Equation (5.12). Now we extend the definition of matrix sine and matrix cosine to any arbitrary square matrix A.

Definition: Let $A = UEV^T$ be an SVD decomposition of A. We define:

$$\begin{aligned} \exp(UEV^T) &= U \cdot \exp(E) \cdot V^T \\ \operatorname{cosm}(UEV^T) &= U \cdot \operatorname{cosm}(E) \cdot V^T \\ \operatorname{sinm}(UEV^T) &= U \cdot \operatorname{sinm}(E) \cdot V^T \\ \operatorname{acosm}(UEV^T) &= U \cdot \operatorname{acosm}(E) \cdot V^T \\ \operatorname{asinm}(UEV^T) &= U \cdot \operatorname{asinm}(E) \cdot V^T \end{aligned} \tag{5.15}$$

Now we are ready to rewrite the geometric distance on $\mathcal{G}r_{\mathrm{n,p}}$, given in Equation (5.10), using the definitions of the matrix arc-cosine. We summarize the result in the following lemma:

Lemma 5.6.1. *Let $\lceil \mathbf{Y}_0 \rfloor, \lceil P \cdot \mathbf{Y}_\tau \rfloor \in \mathcal{G}r_{n,p}$ be two points on the Grassmann manifold. The geometric Grassmannian distance can be rewritten as:*

$$\mathrm{d}_{\mathcal{G}r_{n,p}}\Big(\lceil \mathbf{Y}_0 \rfloor, \lceil P \cdot \mathbf{Y}_\tau \rfloor\Big) = \sqrt{\operatorname{trace}\left(\operatorname{acosm}^T(\mathbf{Y}_0^T P \mathbf{Y}_\tau) \cdot \operatorname{acosm}(\mathbf{Y}_0^T P \mathbf{Y}_\tau)\right)} \tag{5.16}$$

Proof. From Equation (5.9) we have:

$$\mathbf{Y}_0^T P \mathbf{Y}_\tau = V_0 C_\tau V_\tau^T \tag{5.17}$$

Applying $\operatorname{acosm}(\cdot)$ to the equation and using the property in Equation (5.15) we obtain:

$$\operatorname{acosm}(\mathbf{Y}_0^T P \mathbf{Y}_\tau) = V_0 E_\tau V_\tau^T \tag{5.18}$$

But $\operatorname{trace}(ABC) = \operatorname{trace}(BCA)$, so we get the desired result. □

Now we go back to the minimization problem given in Equation (5.4). We redefine it using the expression given in Equation (5.16) for the Grassmannian distance. Let $\mathcal{P}_{\mathrm{n}}\lceil \mathbf{Y}_0 \rfloor_{\mathcal{O}_{\mathrm{p}}}$ and $\mathcal{P}_{\mathrm{n}}\lceil \mathbf{Y}_\tau \rfloor_{\mathcal{O}_{\mathrm{p}}}$ be two points in $\mathcal{GS}_{\mathrm{n,p}}$. Consider the following cost function:

$$J(P) \triangleq \operatorname{trace}\Big(\operatorname{acosm}^T(\mathbf{Y}_0^T P \mathbf{Y}_\tau) \cdot \operatorname{acosm}(\mathbf{Y}_0^T P \mathbf{Y}_\tau)\Big) = \operatorname{trace}(E_\tau^2) \tag{5.19}$$

and consider the following optimization:

$$P = \underset{P \in \mathcal{P}_{\mathrm{n}}}{\arg\min}\, J(P) \tag{5.20}$$

Then clearly this optimization is equivalent to the one given in Equation (5.4).

Now consider the fact that permutation matrices are square matrices obtained by permuting rows of the identity matrix. Thus, they are doubly-stochastic and orthogonal at the same time. In fact, it can be shown that any doubly-stochastic orthogonal matrix is a permutation matrix. Since a set of permutations $\mathcal{P}_{\mathrm{n}}$ is a subset of doubly stochastic matrices $\mathcal{D}_{\mathrm{n}}$, we can relax the optimization problem in Equation (5.20) and write:

$$D = \underset{D \in \mathcal{D}_{\mathrm{n}}}{\arg\min}\, J(D) \tag{5.21}$$

In fact, since $\mathcal{P}_{\mathrm{n}} \subset \mathcal{D}_{\mathrm{n}}$, one easily verifies the identity:

$$\min_{D \in \mathcal{D}_{\mathrm{n}}} J(D) \leqslant \min_{P \in \mathcal{P}_{\mathrm{n}}} J(P) \tag{5.22}$$

If the cost function $J(D)$ is concave (or linear), the optimum of the cost function in Equation (5.21) over all $D \in \mathcal{D}_{\mathrm{n}}$ is attained when D is a permutation matrix. This follows from Birkhoff's theorem [56, 75], which says that a set of doubly-stochastic matrices $\mathcal{D}_{\mathrm{n}}$ is the convex hull over a group of permutations $\mathcal{P}_{\mathrm{n}}$ and also from the fact that the concave objective function achieves its minimum at the boundaries. Thus, the inequality in Equation (5.22) is in fact an equality and we can write:

$$P = \underset{D \in \mathcal{D}_{\mathrm{n}}}{\arg\min}\, J(D) \subset \mathcal{P}_{\mathrm{n}} \tag{5.23}$$

In order to minimize an objective function, one typically first looks for its derivatives. The derivative of the true distance function in Equation (5.19) is not easily computable. This is due to the fact that all three matrices in the SVD decomposition, given in Equation (5.9), depend on the permutation matrix P. We were able to take such a derivative only under the assumption that a diagonal matrix E_τ commutes with any orthogonal matrix, which is only the case when all p principal angles are the same. Under this assumption, the derivative is given by:

$$\triangledown J_{Q,\mathcal{GS}_{\mathrm{n,p}}}(D) \approx -2\mathbf{Y}_0 \left(2E_\tau \mathrm{sinm}^{-1}(2E_\tau)\right)^{\frac{1}{2}} \mathbf{Y}_0^T D \mathbf{Y}_\tau \left(2E_\tau \mathrm{sinm}^{-1}(2E_\tau)\right)^{\frac{1}{2}} \mathbf{Y}_\tau^T \tag{5.24}$$

By further simplifying the quadratic case in Equation (5.24) by its linear counterpart, we have the derivative for the linear case to be approximated by:

$$\nabla J_{L,\mathcal{GS}_{\mathrm{n,p}}}(D) \approx -\mathbf{Y}_0 \left(2E_\tau \mathrm{sinm}^{-1}(2E_\tau)\right) \mathbf{Y}_\tau^T \tag{5.25}$$

Both Equation (5.24) and Equation (5.25) are hard to analyze and are not adequate for practical purposes. In the following sections, we give alternative approximations, which are easily solved using concave quadratic programming and linear programming.

5.7 An approximation to the optimization problem via QP

An interesting result can be obtained by using the trigonometric identity:

$$2\mathrm{cosm}^2(E) = I + \mathrm{cosm}(2E) \tag{5.26}$$

From it we get:

$$4E^2 = \mathrm{acosm}^2(2\mathrm{cosm}^2(E) - I) \tag{5.27}$$

Then, we can rewrite the optimization in Equation (5.23) where the cost function is given by Equation (5.19) as follows:

$$\begin{aligned} P &= \arg\min_{D\in\mathcal{D}_{\mathrm{n}}} \mathrm{trace}(E_\tau^2) \\ &= \arg\min_{D\in\mathcal{D}_{\mathrm{n}}} \mathrm{trace}(4E_\tau^2) \\ &= \arg\min_{D\in\mathcal{D}_{\mathrm{n}}} \mathrm{trace}(\mathrm{acosm}^2(2C_\tau^2 - I)) \end{aligned} \tag{5.28}$$

If we assume that minimizing the trace of the square of the matrix arc-cosine is about the same as minimizing the negative trace of its argument, we can approximate:

$$\begin{aligned} P_Q &= \arg\min_{D\in\mathcal{D}_{\mathrm{n}}} -\mathrm{trace}(2C_\tau^2 - I) \\ &= \arg\min_{D\in\mathcal{D}_{\mathrm{n}}} -\mathrm{trace}(C_\tau^T C_\tau) \\ &= \arg\min_{D\in\mathcal{D}_{\mathrm{n}}} -\mathrm{trace}((V_0^T \mathbf{Y}_0^T P \mathbf{Y}_\tau V_\tau)^T (V_0^T \mathbf{Y}_0^T P \mathbf{Y}_\tau V_\tau)) \\ &= \arg\min_{D\in\mathcal{D}_{\mathrm{n}}} -\mathrm{trace}(P^T \mathbf{Y}_0 \mathbf{Y}_0^T P \mathbf{Y}_\tau \mathbf{Y}_\tau^T) \end{aligned} \tag{5.29}$$

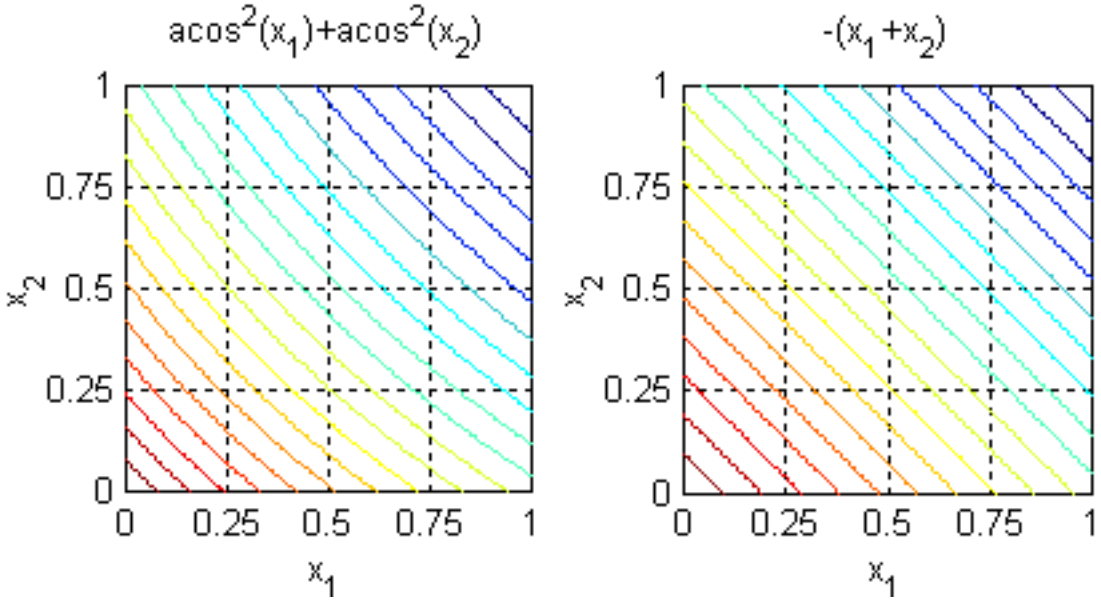

Fig. 5.3: Approximating $\text{trace}(\text{acosm}^2(X))$ by $-\text{trace}(X)$ for diagonal X

In Figure 5.3 we show both cost functions for the $p = 2$ case. On the left-hand side, we provide a contour plot of the $f(X) = \text{trace}(\text{acosm}^2(X))$, and on the right-hand side we provide a contour plot of the $g(X) = -\text{trace}(X)$. Except for a small curvature, $f(X)$ and $g(X)$ behave about the same. So, the assumption we made seems to be a good approximation. Then, let us define the quadratic cost function to be:

$$J_Q(D) = -\text{trace}(D^T \mathbf{Y}_0 \mathbf{Y}_0^T D \mathbf{Y}_\tau \mathbf{Y}_\tau^T) \tag{5.30}$$

Its gradient, using standard tools of matrix calculus [42], is easily computed to be:

$$\nabla J_Q(D) = -2(\mathbf{Y}_0 \mathbf{Y}_0^T) D (\mathbf{Y}_\tau \mathbf{Y}_\tau^T) \tag{5.31}$$

Now we rewrite the optimization problem in a canonical form, so that it can be solved numerically using standard quadratic programming routines. We apply the $\text{vec}(\cdot)$ notation, defined in [42, 76], to convert matrices into vectors. Let $d = \text{vec}(D)$. Then using the rule $\text{vec}(AXB) = (B^T \otimes A) \cdot \text{vec}(X)$ we obtain:

$$\nabla J_Q(d) = -2(\mathbf{Y}_0 \mathbf{Y}_0^T \otimes \mathbf{Y}_\tau \mathbf{Y}_\tau^T) \cdot d \tag{5.32}$$

Then we recover the canonical cost function in the vec$(\cdot)$ notation to be:

$$J_Q(d) = -d^T \cdot (\mathbf{Y}_0\mathbf{Y}_0^T \otimes \mathbf{Y}_\tau\mathbf{Y}_\tau^T) \cdot d \tag{5.33}$$

This is the canonical form for defining a quadratic function for the optimization routines. The quadratic programming routines typically require an initial guess to be passed in. They also require a set of constrains. In the following section, we derive an even coarser linear approximation. It can be used as an initial guess to the quadratic optimization. We will also derive a set of constrains that can be used for both quadratic and linear cost functions. They are based on the properties of permutation matrices.

5.8 An approximation to the optimization problem via LP

Consider the optimization problem, given in Equation (5.4). The associated cost function is given by Equation (5.19). As we already discussed, optimization over the true geometric cost function is computationally very demanding and is not plausible. If we approximate the distance in Equation (5.4) by the Euclidean distance, then the problem becomes:

$$P_L = \underset{D\in\mathcal{D}_n}{\arg\min} \|\mathbf{Y}_0 - D\mathbf{Y}_\tau\|_F \tag{5.34}$$

This optimization problem in Euclidean space is also known as the correspondence problem, and it is easily converted into a linear optimization framework, [73, 72]:

$$\begin{aligned} P_L &= \underset{D\in\mathcal{D}_n}{\arg\min} \|\mathbf{Y}_0 - D\mathbf{Y}_\tau\|_F \\ &= \underset{D\in\mathcal{D}_n}{\arg\min} \operatorname{trace} (\mathbf{Y}_0 - D\mathbf{Y}_\tau)^T (\mathbf{Y}_0 - D\mathbf{Y}_\tau) \\ &= \underset{D\in\mathcal{D}_n}{\arg\min} \operatorname{trace} \left(\mathbf{Y}_0^T\mathbf{Y}_0 - 2\mathbf{Y}_0^T D\mathbf{Y}_\tau + \mathbf{Y}_\tau^T\mathbf{Y}_\tau\right) \\ &= \underset{D\in\mathcal{D}_n}{\arg\min} -\operatorname{trace} \left(\mathbf{Y}_0^T P\mathbf{Y}_\tau\right) \end{aligned} \tag{5.35}$$

Then, let us define the linear cost function to be:

$$J_L(D) = -\operatorname{trace}(\mathbf{Y}_0^T D\mathbf{Y}_\tau) = -\operatorname{trace}(D \cdot \mathbf{Y}_\tau\mathbf{Y}_0^T) \tag{5.36}$$

Its gradient, using standard tools of matrix calculus [42], is easily computed to be:

$$\triangledown J_L(D) = -\mathbf{Y}_0\mathbf{Y}_\tau^T \tag{5.37}$$

Now we rewrite the optimization problem in a canonical form. For this, as before, we apply the vec$(\cdot)$ notation, defined in [42, 76]. Let $d = \mathrm{vec}(D)$. Then we obtain:

$$\triangledown J_L(d) = -\mathrm{vec}(\mathbf{Y}_0 \mathbf{Y}_\tau^T) \tag{5.38}$$

This leads to a canonical linear cost function

$$J_L(d) = -\mathrm{vec}(\mathbf{Y}_0 \mathbf{Y}_\tau^T)^T \cdot d \tag{5.39}$$

This is the canonical form for defining a linear function for the optimization routines.

We note that the constraints on D to be a doubly-stochastic matrix are expressed as:

$$D \cdot \mathbf{1} = \mathbf{1} \quad , \quad \mathbf{1}^T \cdot D = \mathbf{1}^T \quad , \quad 0 \leq d_{i,j} \tag{5.40}$$

Now we rewrite the constrains in a canonical form, using the vec$(\cdot)$ notation. The canonical linear constraints are given by:

$$\begin{aligned} (\mathbf{1}^T \otimes I) \cdot d &= \mathbf{1} \\ (I \otimes \mathbf{1}^T) \cdot d &= \mathbf{1} \\ \mathbf{0} &\leq d \end{aligned} \tag{5.41}$$

The problem defined by the linear cost function Equation (5.39) along with the constrains defined by Equation (5.41) is the standard linear optimization problem and can be solved by standard simplex or simplex-like algorithms. Faster implementations of simplex do exist specifically for optimizations over the Birkhoff's polytope.

Also, the solution found by the linear cost function in Equation (5.39) can be fed into the quadratic cost function in Equation (5.33) along with the constrains defined by Equation (5.41) is order to perform the standard quadratic optimization problem.

Even though we derived both linear approximation and more precise quadratic approximation, in the following sections we will give a conjecture that states that linear programming leads to sufficiently precise results for defining a local geometry in the shape space $\mathcal{GS}_{\mathrm{np}}$. Thus, we are going to use only the linear approximation for the experimental results in Chapter 6.

5.9 Canonical representation of a set and local geometry of $\mathcal{GS}_{np}$

Suppose we are given a set of points on the Grassmann manifold $\mathcal{G}r_{\mathrm{n}\mathrm{p}}$. These points give a representation of shapes that achieves affine-invariance only. In order to obtain a representation that achieves affine-permutation-invariance, we solve the permutation invariance for the set by finding a canonical member for each member of the set. We proceed as follows. Pick a point that lies sufficiently close to all other points in the set in terms of the similarity defined on $\mathcal{GS}_{\mathrm{n}\mathrm{p}}$, and call it a *"reference point"*. For example, mean of this set might be known a-priory. It would be a good reference point. If mean is not known, one could randomly pick one of the members to be the reference point. Next, solve an optimization between the reference point and each member of the set. It is given in Equation (5.23), where we use the linear cost function, given in Equation (5.39). Assign an output of the optimization — the optimal permutation matrix — to each member of the set as its label. This is an algorithm for labeling points on the Grassmann manifold with permutation matrices as their labels.

In order to obtain a canonical representation of the set, apply the inverse permutations to the ones stored in the label of each point. This is an alignment operation that brings all points of the set in correspondence with each other, and with the reference point.

Now we give two conjectures that allow us to establish a local geometric structure in the shape space $\mathcal{GS}_{\mathrm{n}\mathrm{p}}$.

The first conjecture states that if the minimized distance found as a result of linear programming is large, then the true minimized distance would have been large as well. In other words, it tells us that the linear programming solution is a mapping that preserves distances locally. For this reason, given two shapes, we can establish how similar the shapes are, and if they are sufficiently similar, we can compute their distance.

In addition, we establish another conjecture that tells us that a constellation of similar shapes will get mapped to the same local neighborhood of the Grassmann manifold via the algorithm for canonical representation of a set, described above. Thus, for sufficiently similar shapes we not only can compute distances, but also perform other geometric concepts

derived for the Grassmann manifold. In particular, we can compute a sample mean and higher order moments of a constellation of shapes that are sufficiently similar to their sample mean.

Conjecture 5.9.1 (LP is sufficient). *Let $\lceil \mathbf{Y}_0 \rfloor, \lceil \mathbf{Y}_\tau \rfloor \in \mathcal{G}r_{n,p}$ be two points on the Grassmann manifold with the Grassmanian distance between them being possibly large. If the true unknown distance, as given by Equation (5.4) is sufficiently small, then the linear programming will succeed, finding the permutation P_L such that the Grassmanian distance between the permuted shape $\lceil \mathbf{Y}_0 \rfloor, \lceil \mathrm{P_L} \cdot \mathbf{Y}_\tau \rfloor \in \mathcal{G}r_{n,p}$ will be small.*

Conjecture 5.9.2 (local geometry of $\mathcal{GS}_{\mathbf{n,p}}$). *Canonical representation maps similar points into the same neighborhood of the Grassmann manifold.*

The extensive experimental results show that both conjectures are surely true. For example, consider the following experiment. We take some random point on the Grassmann manifold and fix it to be the reference point for the canonical representation algorithm. Then we take a set of random directions, and move from the reference point in these directions by small, in terms of the distance on the Grassmann manifold, random amounts. Thus, we obtain a constellation of points on the Grassmann manifold that lies sufficiently close to the reference point. By running the canonical representation algorithm on this set we expect each member of the constellation to get labeled by the identity permutation. In other words, the identity permutation should be the optimal alignment permutation for each member of this set. Now we apply random permutations to each member of the set. This makes members of the constellation to get spread far away from the reference point, in terms of geometry of the Grassmann manifold. Now we run the canonical representation algorithm and verify the recovered optimal permutations. These should be the inverses to the random permutations that were applied to each of the members. Experiments show that for a sufficiently dense constellations we do recover the correct permutations, using the linear programming algorithm. Typical values for the radius of the constellation in order to achieve the reconstruction are about $0.05 - 0.1$ on a scale 0 to 1. This confirms the first conjecture.

In order to verify the second conjecture, we take each un-aligned member of the constellation and connect it by the Grassmannian minimal geodesic segment to the reference point. This geodesic segment might be quite long, since the members are not aligned. We densely sample the geodesic segment and obtain points along it, starting from the reference point and ending at the selected un-aligned member of the constellation. Then we perform the canonical representation algorithm between the reference point and each sampled point along the geodesic. Each sampled point gets labeled with the optimal permutation. The experiment is repeated sufficiently many times. The obtained results are very interesting. Sampled points along the geodesic that lie sufficiently close to the reference point will always get labeled with the identity permutation. Also, sampled points that lie sufficiently close to the endpoint, which is the selected un-aligned member of the constellation, will always get labeled with exactly the same permutation that is stored in the label of the un-aligned member. The experiment verifies the second conjecture. Sampled points that are far away from both the reference point and the endpoint will get labeled with various permutations. For these points, linear programming fails to find correct permutations. Thus, the true distance is large, and the experimentally measured distance is even larger. For this reason, we can only say that these shapes are not similar to the reference point. Thus, we have a valid similarity measure, which is exact for similar shapes and coincides with the Grassmannian distance.

Now we summarize the results obtained in this chapter.

5.10 Summary

In this chapter we established a representation for the affine-permutation-invariant shape space. This quotient space is not a Riemannian manifold. For this reason we presented a similarity measure that is invariant to both affine transformations and permutations as a solution to the optimization problem. As the true distance minimization is combinatorially complex, we relaxed an optimization problem to doubly-stochastic matrices and presented an approximate solution using linear programming and quadratic programming.

We established a conjecture that the affine-permutation-invariant shape space locally looks like the neighborhood of the real Grassmann manifold $\mathcal{G}r_{\mathfrak{n}\mathfrak{p}}$. We gave an analytical mapping that maps neighborhoods in the shape space to the neighborhoods of the real Grassmann manifold. This map is the solution to the distance minimization problem using linear programming. For this reason, given two shapes, we can establish how similar the shapes are, and if they are sufficiently similar, we can compute their distance as well as perform other geometric concepts presented for the Grassmann manifold. In particular, we can compute the sample mean and higher order moments of a constellation of shapes that are sufficiently similar to their sample mean.

6. CASE STUDIES: APPLICATIONS IN SHAPE REPRESENTATION

6.1 Preliminaries

Object recognition demands a measure of similarity of two shapes. But what is the metric? In fact, there are many metrics one could consider. Some of the most suggestive are those derived from Riemannian metrics, [81]. In this chapter, we will present some experiments that will show that the metric we chose in fact does achieve very good results.

First we start by giving some general comments. Visualization of 2D shapes is easy. One simply plots feature points in the image plane. These visualizations are defined only up to an arbitrary affine transformation. In other words, numerically shapes are represented by an orthonormal $n \times p$ matrices, which can be visualized in many different ways. Thus, whenever we plot some view of a shape, we can choose any arbitrary affine transformation and apply to it before plotting it.

Visualization of 3D shapes is trickier. For these purposes one could use, for example, the package DistMesh introduced by [94]. One could also use the package in order to take 3D objects and triangulate them to get vertex nodes that can be used as feature points.

The algorithms developed require the number of feature points n to be the same. Our experimental results show that the distances and the geodesics computed are very insensitive to the number of points chosen, as long as it is high enough to represent shapes accurately. For this reason, if one needs to compare shapes that have different number of points, one could either randomly drop some points, or add some points, possibly by resampling the shape, or maybe simply by repeating the same points.

Now we present some experimental results.

6.2 Application: Multi-author handwritten digit classification on $\mathcal{G}r_{n,p}$

Here we present an application of the derived formulas to the classification problem.

We used the database of pen-based handwritten digits 'pendigits' from the UCI Repository of Machine Learning Databases, [83]. The source of this database is [3, 4]. This database consists of 250 samples from each of 44 different writers, for a total of $10,992$ good samples. There are total of 10 classes — digits 0 though 9. The database is split into three parts. The samples written by first 30 writers is split into two parts: a training set consisting of $3,745$ samples and a writer-dependent test consisting of $3,749$. The digits written by the other 14 writers is used for writer-independent testing.

WACOM PL-100V pressure sensitive tablet with an integrated LCD display and a cordless stylus was used for capturing the data. The input and display areas are located in the same place. Attached to the serial port of an Intel 486 based PC, it allows to collect handwriting samples. The tablet sends x and y tablet coordinates and pressure level values of the pen at fixed time intervals (sampling rate) of 100 miliseconds.

These writers are asked to write 250 digits in random order inside boxes of 500 by 500 tablet pixel resolution. Subjects are monitored only during the first entry screens. Each screen contains five boxes with the digits to be written displayed above. Subjects are told to write only inside these boxes. If they make a mistake or are unhappy with their writing, they are instructed to clear the content of a box by using an on-screen button. The first ten digits are ignored because most writers are not familiar with this type of input devices, but subjects are not aware of this.

For the purposes of this application, following [3, 4], we use only (x, y) coordinate information. The stylus pressure level values are ignored. The raw data that we capture from the tablet consist of integer coordinates between 0 and 500 (tablet input box resolution), captured as a temporal sequence of the movements of the pen tip. Handwriting can be represented as a time-ordered sequence of coordinates of the pen tip on the tablet with varying speed and pressure at points. The digits were written by people having various styles of writing. The raw data contains enormous variations in writing speed. Also, number of

temporal samples per digit varies drastically. Its typical range is in between $10 - 30$. Some preprocessing steps are needed in order to normalize time variations and variations in the amount of samples.

We fix a value of n to be the desired number of samples per digit, and we perform spatial resampling by placing points regularly spaced along the trajectory arc. A typical value of n that we used was $n = 100$. We tried both increasing and decreseaing the value of n, but it has very little effect to the performance of the classifier. Also, feature points are ordered in time, so they can be assumed to be labeled. The MATLAB code for resempling is presented below:

```
function [X,pen] = loadDigit(n,idx)

global pens;
pen = pens.digits(idx);

x = pen.data(:,1);
y = pen.data(:,2);

dx=diff(x);
dy=diff(y);
dt=sqrt(dx.^2+dy.^2);

I=find(dt~=0);
dx=dx(I);
dy=dy(I);
dt=dt(I);

x = x(1) + [0;cumsum(dx)];
y = y(1) + [0;cumsum(dy)];
t = 0    + [0;cumsum(dt)];
tt = linspace(0,max(t),n);

X = interp1(t,[x y],tt,'pchip');
```

This preprocessing leads to the configuration matrix $\mathbf{X} \in \mathcal{SC}_{n,p}$ with $p = 2$, see Figure 2.2.

The nearest neighbor rule is a simple and powerful classification technique. In the basic nearest neighbor classifier, each training sample is used as a prototype, and a test sample is assigned to the class of the closest prototype. When the number of prototypes is large,

this method requires a large memory space and long computing time, since each of the test samples has to be compared with each of the prototypes.

For the purposes of this demonstration, we propose a preprocessing step based on sample mean calculation, which takes training set as an input, and produces a reduced set of prototypes. This preprocessing step is not meant to increase the quality of the nearest neighbor qualifier. It is used purely for the demonstration of calculation of sample mean of a large number of samples. Also, this preprocessing algorithm can be used to visualize the dataset in a compact way, as we will see later.

As an output of this algorithm, we produce a reduced set of prototypes, where each prototype is a sample mean of samples from the training set in its proximity. Recall that the normalized distance on the Grassmann manifold, as given in Equation (4.32), takes values in the range $[0, 1]$. To control the algorithm for producing a reduced set of prototypes, we pick some threshold λ value based on which we determine if samples in the training set are nearby to each other. Some typical value for the threshold that we have considered are $\lambda = 0.05, 0.15, 0.3$. The algorithm for generating prototypes is as follows. We take some random sample of the training set as an initial reference point. We find all other samples in the training set that lie closer than the threshold λ to the initial reference point. Then we compute a sample mean of these samples, and take newly computed sample mean as initial reference point. We repeat the step of finding samples that lie closer than the threshold λ to the initial reference point. The process is repeated a few times, typically 2 or 3 times until certain exit conditions are met. The output of this iteration is a cluster of samples that lie within the distance λ apart from their sample mean. We remove this cluster of samples from the training set, and take their mean to be one of the prototypes for the classification. Then we repeat the process again and again till no more samples are left in the training set. Clearly, if one takes threshold value to be $\lambda = 0$, then each prototype will be a sample mean of exactly one sample from the training set, reducing down to a conventional nearest neighbor classification. By taking λ large, the number of prototypes will get reduced, thus causing poor classification performance.

Now we present some results with different values of λ. In this particular example, we take $n = 100$, and $p = 2$ since digits are 2D shapes. The value of n does not play any major role. In fact, we will see later that shapes are resilient to the dimensionality n. Thus, it can be any sufficiently large number. In Figure 6.1 we present class separation in between different digits. We remind the reader that the training set consists of $3,745$ samples. With the value of $\lambda = 0.3$ our algorithm produces only 109 prototypes. For $\lambda = 0.15$ there are 461 prototypes. Finally, for the value of $\lambda = 0.05$ there are $2,877$ prototypes, which is essentially performing a full nearest neighbor classification. We also compute class separation in between different classes. This is essentially a distance in between two closest prototypes belonging to the two classes under the considerations. These results are presented in the matrices in Figure 6.1.

Figure 6.2 visualizes all 109 prototypes produced from the training set of $3,745$ samples for the value of $\lambda = 0.3$. Above each prototype, we display number of samples that were used in its calculation. Recall that each prototype is simply a sample mean of all of its members. First of all, this figure shows practical ability of computing approximate expected values with high accuracy even for a widely spread constellation of points of a large size. For example, consider the digit '2' displayed at the intersection of forth column and third row. It consists of 328 members. We do not display the histogram of distances from it to each of its members, but it is approximately uniform with distance spread between 0 and $\lambda = 0.3$. The value of normalized distance 0.3 is quite large. Also, the number of samples is large as well. Yet, our algorithm allows a very efficient and linear calculation of the sample mean that is approximately correct with an error being very small, negligible from the practical viewpoint. This example verifies our algorithm for the barycentric computation.

Let us look again at Figure 6.2. We see that our clustering algorithm creates large clusters for each digit at first, and then small clusters of digits that are either outliers, or are not modeled well by an affine distortions. For example, let us look at the intersection of ninth column and third row. It is a digit '2' that is improperly labeled as '3' in the training set. Our algorithm was able to correctly detect this outlier. Also, one could consider all

clusters consisting of a few digits to be outliers, even if they are labeled correctly. There are samples in the training set that were written by sloppy writers, and most likely should be omitted from the training set. For this demonstration, we simply leave the outliers in the set of prototypes.

For $\lambda = 0.15$ there are 461 prototypes, out of which 79 belong to the digits with label '1'. These are displayed in Figure 6.3. Again, we see that the first cluster contains most of the training set.

Finally, in Figure 6.4 we show confusion matrices and error rates for the classification. As expected, for $\lambda = 0.3$ classification has 17% error rate. This is not surprising since we are running the nearest neighbor classifier against the set of prototypes consisting of only 109 prototypes. In a way, this shows that this set is not large enough to train the classifier. For $\lambda = 0.05$. classification achieves a very good error rate. For the writer-dependent set, it is only 3.2%, and for writer-independent set, it is only 5.9%, and overall error rate is only 3.1%. These results are comparable to the error rates reported on the same dataset by [3, 4]. Once again, we are only showing a prototype implementation of the classifier. One could achieve even better rates by removing outliers form the training set, or by implementing a better classifier, or even by combining multiple classifiers in order to boost classification accuracy, as it is done in [3, 4]. For example, one could use a classifier on the Grassmann manifold combined with a classifier that uses a different space for the features. By analyzing the confusion matrix, we see that digits '6' and '9' get confused a lot. This is not surprising, since we are allowing arbitrary affine transformations, including a rotation by 180 degrees, which makes them look very similar to each other on the shape space.

To summarize, with this real-world application we showed that the metric we chose does in fact achieve good results.

training set : 3,745

thres=0.30 # of prototypes: 109

	1	2	3	4	5	6	7	8	9	0
1	.000	.214	.152	.276	.217	.227	.079	.312	.162	.282
2	.214	.000	.194	.169	.082	.324	.293	.308	.241	.504
3	.152	.194	.000	.154	.122	.291	.178	.402	.263	.247
4	.276	.169	.154	.000	.147	.215	.182	.352	.288	.283
5	.217	.082	.122	.147	.000	.173	.337	.247	.137	.216
6	.227	.324	.291	.215	.173	.000	.201	.318	.164	.298
7	.079	.293	.178	.182	.337	.201	.000	.226	.247	.339
8	.312	.308	.402	.352	.247	.318	.226	.000	.269	.273
9	.162	.241	.263	.288	.137	.164	.247	.269	.000	.273
0	.282	.504	.247	.283	.216	.298	.339	.273	.273	.000

thres=0.15 # of prototypes: 461

	1	2	3	4	5	6	7	8	9	0
1	.000	.176	.152	.143	.182	.227	.061	.258	.182	.212
2	.176	.000	.074	.106	.088	.155	.132	.190	.138	.262
3	.152	.074	.000	.130	.114	.240	.157	.288	.232	.159
4	.143	.106	.130	.000	.177	.147	.136	.277	.190	.165
5	.182	.088	.114	.177	.000	.107	.236	.173	.075	.153
6	.227	.155	.240	.147	.107	.000	.145	.280	.055	.095
7	.061	.132	.157	.136	.236	.145	.000	.114	.169	.204
8	.258	.190	.288	.277	.173	.280	.114	.000	.182	.219
9	.182	.138	.232	.190	.075	.055	.169	.182	.000	.168
0	.212	.262	.159	.165	.153	.095	.204	.219	.168	.000

thres=0.05 # of prototypes: 2,877

	1	2	3	4	5	6	7	8	9	0
1	.000	.113	.083	.128	.153	.156	.036	.236	.150	.134
2	.113	.000	.024	.064	.047	.089	.077	.190	.104	.238
3	.083	.024	.000	.105	.104	.215	.103	.247	.147	.114
4	.128	.064	.105	.000	.092	.079	.073	.240	.149	.160
5	.153	.047	.104	.092	.000	.092	.176	.147	.067	.108
6	.156	.089	.215	.079	.092	.000	.095	.257	.035	.081
7	.036	.077	.103	.073	.176	.095	.000	.105	.116	.135
8	.236	.190	.247	.240	.147	.257	.105	.000	.155	.226
9	.150	.104	.147	.149	.067	.035	.116	.155	.000	.088
0	.134	.238	.114	.160	.108	.081	.135	.226	.088	.000

Fig. 6.1: Class separation in the $\mathcal{G}r_{\mathrm{np}}$ space

Fig. 6.2: Prototypes for $\lambda = 0.3$ and classes 0 though 9

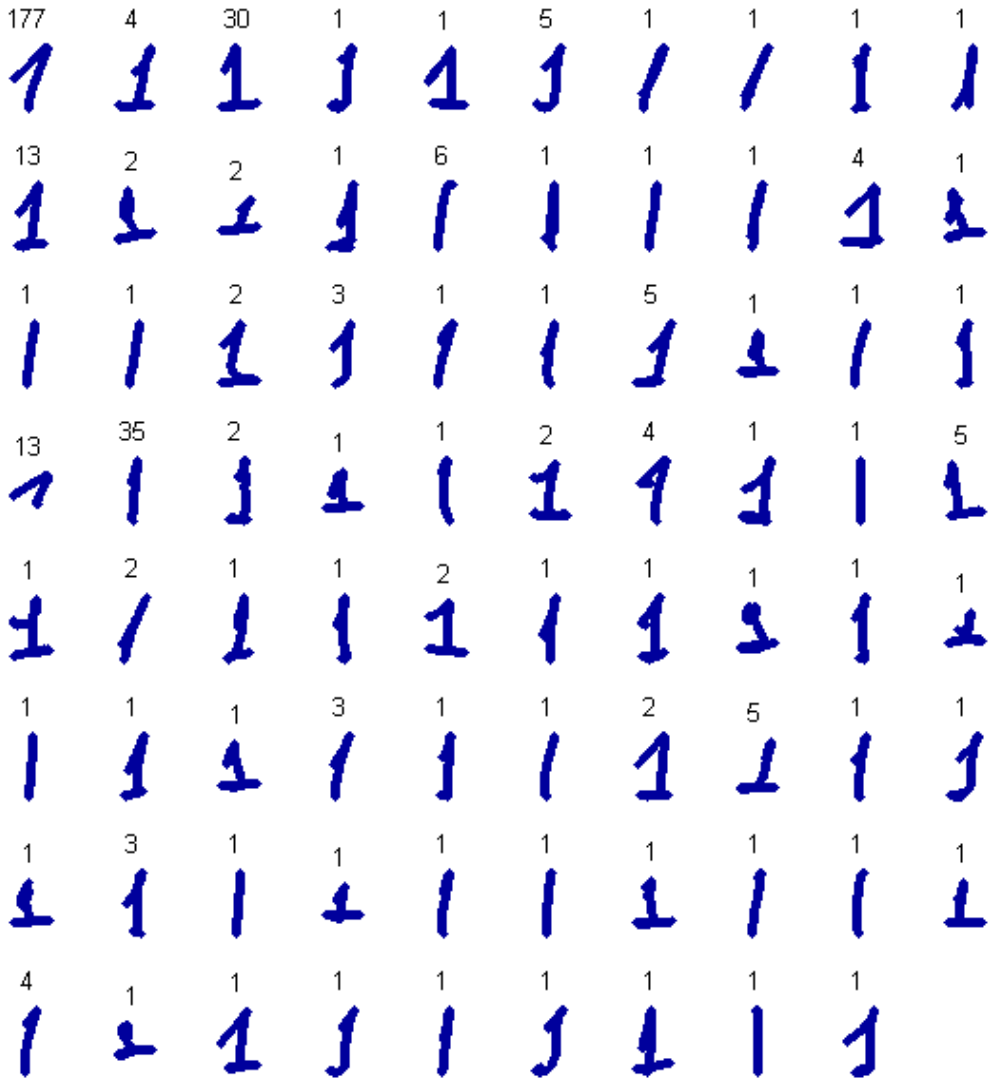

Fig. 6.3: Prototypes for $\lambda = 0.15$ and class 1

training set : 3,745
writer-dependent set : 3,749
writer-independent set : 3,498
total: 10,992

thres=0.30 **# of prototypes: 109**

	1	2	3	4	5	6	7	8	9	0
1	931	25	37	2	2	1	140	1	4	0
2	10	731	117	123	148	0	12	0	3	0
3	19	0	981	1	45	0	6	0	3	0
4	6	164	8	906	22	8	24	0	3	3
5	11	47	28	2	851	64	0	6	35	11
6	14	0	0	5	0	769	1	1	260	6
7	31	10	6	31	0	50	969	8	36	1
8	3	5	0	0	2	1	39	994	9	2
9	11	4	6	8	157	15	4	4	843	3
0	0	0	0	0	2	1	2	1	1	1136

errors in training set	563	15%
errors in writer-dependent	634	17%
errors in writer-independent	684	20%
total errors	1881	17%

thres=0.05 **# of prototypes: 2,877**

	1	2	3	4	5	6	7	8	9	0
1	1069	16	4	1	2	1	40	0	10	0
2	1	1123	3	4	10	0	2	0	1	0
3	4	0	1048	0	0	0	1	0	2	0
4	0	28	1	1096	2	0	17	0	0	0
5	2	12	5	1	1028	2	0	1	4	0
6	1	3	0	2	0	997	0	0	52	1
7	23	7	1	7	0	16	1085	2	1	0
8	1	1	0	0	1	0	10	1041	1	0
9	2	1	6	2	5	3	3	4	1029	0
0	0	0	0	0	0	1	0	1	1	1140

errors in training set	8	0.2%
errors in writer-dependent	120	3.2%
errors in writer-independent	208	5.9%
total errors	336	3.1%

Fig. 6.4: Confusion matrix for classification results

6.3 Application: Geodesic triangle on $\mathcal{G}r_{n,p}$

Consider and example given in Figure 6.5. We take three different shapes $\lceil \mathbf{Y}_1 \rfloor, \lceil \mathbf{Y}_2 \rfloor, \lceil \mathbf{Y}_3 \rfloor \in \mathcal{G}r_{100,2}$. In this particular example we took digits '3', '2', and '5' from the same multi-author hand-written digit database, described in Section 6.2. The normalized distances in between them are quite large and asymetric:

$$\mathrm{d}_n\big(\lceil \mathbf{Y}_1 \rfloor, \lceil \mathbf{Y}_2 \rfloor\big) = 0.6772$$
$$\mathrm{d}_n\big(\lceil \mathbf{Y}_1 \rfloor, \lceil \mathbf{Y}_3 \rfloor\big) = 0.6730$$
$$\mathrm{d}_n\big(\lceil \mathbf{Y}_2 \rfloor, \lceil \mathbf{Y}_3 \rfloor\big) = 0.1647$$

For planar triangles, the center of mass of a triangle is a third of the way from the base. The centroid of a triangle is at the point that is at the intersection of: the three side bisectors; the three area bisectors (which are the side bisectors); and the three lines one third of the way up from the three bases. If the triangle only has three equal point masses on its vertices the center of mass lands on the same place. Thus, in order to find the center of mass one needs to first find the three side bisectors, and then find their intersection point a third of the way up from the three bases.

For the geodesic triangles, the three side bisectors will not intersect exactly at one point. But they will intersect approximately at one point — the center of mass. This is shown in Figure 6.5. In fact, there are three ways of finding the center of mass. First, we find the midpoint of one of its sides, and then we move $\frac{1}{3}$ of the way up from this point toward the third vertex. We compute all three ways of finding an approximate center of mass, and we calculate the error of our approximation. It is only:

$$\epsilon = 0.0047 \tag{6.1}$$

which is rather small compared to the distances between the three points. This experiment verifies that for most practical applications this error should be considered negligible.

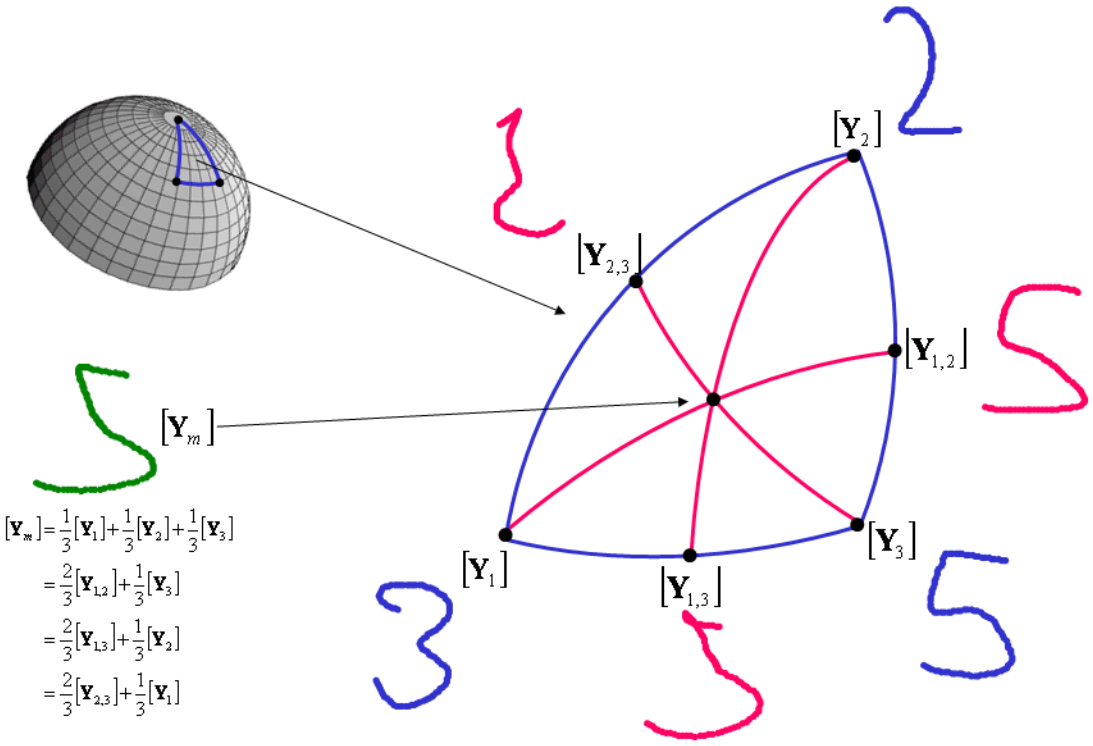

Fig. 6.5: Geodesic Triangle: medians intersect at the barycenter of 3 points

6.4 Application: Shape morphing and geodesics on $\mathcal{GS}_{n,p}$

Now we will present some results on computing distances and morphing shapes in the shape space $\mathcal{GS}_{n,p}$.

In Figure 6.6 we present a geodesic connecting two shapes that lie in the shape space $\mathcal{GS}_{210,2}$. We pick our initial shape to be a contour of the triangle, and our final shape to be a contour of the circle. Then, using the canonical representation algorithm, described in Section 5.9, we align the shapes, mapping the back to the Grassman manifold $\mathcal{G}r_{210,2}$. Then using Equation (4.25) we determine the distance in between the two shapes. If we normalize the distance to be in the range $[0, \ldots, 1]$, then the normalized distance for this example is $d = 0.164$, as shown underneath the figure. Also, using Equation (4.56), we can obtain other shapes that lie on the geodesic given by these initial and final shapes. In this example, we divide the geodesic into 9 equal parts, and compute intermediate shapes along the geodesic. The distance between adjacent shapes will be $d/9$.

Another example is given in Figure 6.7. In this example, our initial shape is a 2D face template, and our final shape is a bicycle. These shapes lie on the $\mathcal{GS}_{2000,2}$. The normalized distance between the shapes is $d = 0.683$. Thus, these shapes lie far from each other. As before, we show some intermediate shapes as well. They show what changes a shape of the face undergoes to become the shape of a bicycle.

Yet another example is given in Figure 6.8. Here we are considering 3D shapes. In this

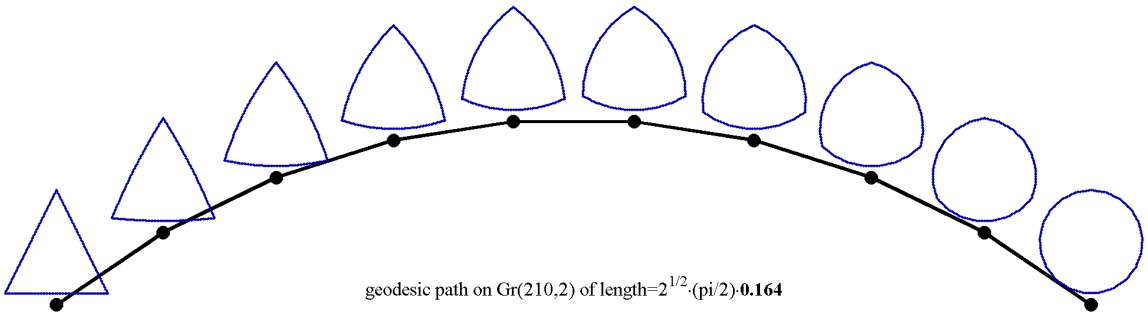

Fig. 6.6: Morphing the contour of a triangle onto the contour of a circle

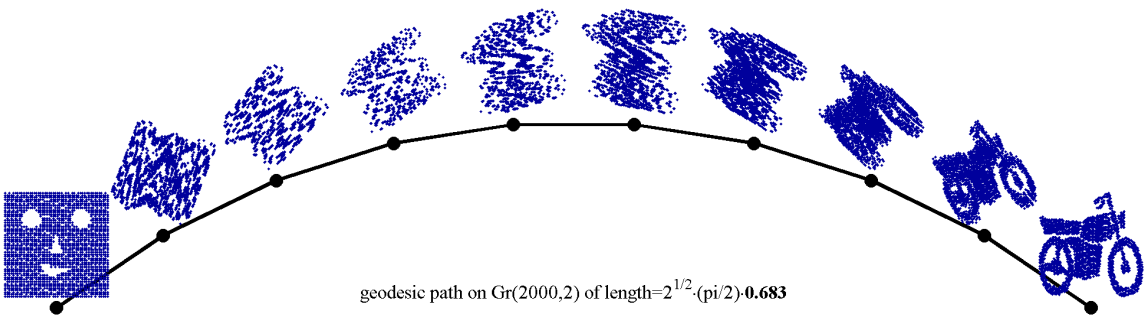

Fig. 6.7: Morphing a face template onto a bicycle

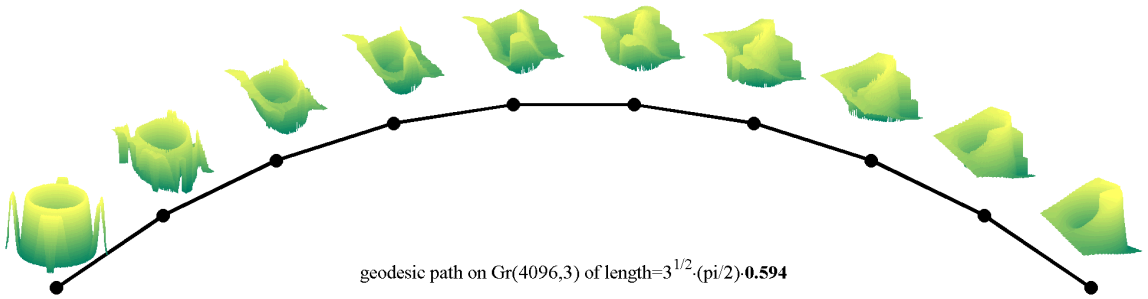

Fig. 6.8: Morphing a fractal bowl onto a Moebius strip

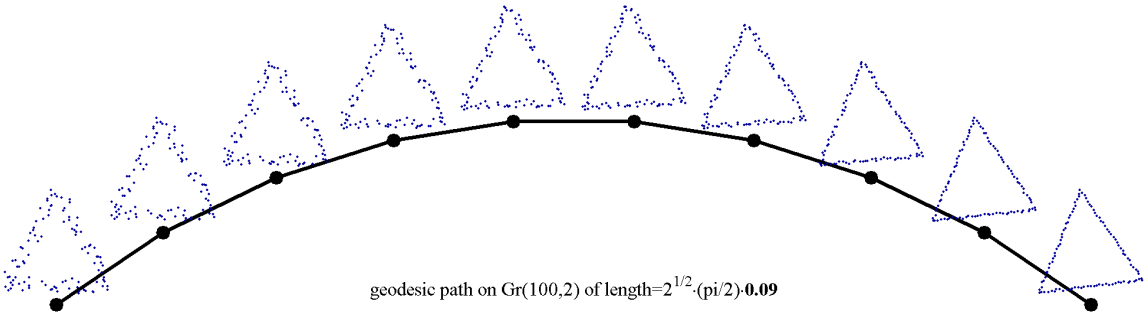

Fig. 6.9: Morphing one distorted triangle onto the other

example, our initial shape is a 3D fractal bowl, and our final shape is a Moebius strip. These shapes lie on the $\mathcal{GS}_{4096,3}$. The normalized distance in between the shapes is $d = 0.594$.

The last example we consider is given in Figure 6.9. In this example, we take the contour of a triangle, and we make two distorted shapes out of it. First, we apply two random affine distortions to obtain two different shapes. Then, we add additive Gaussian noise to the location of the feature points in both shapes. In the first shape we add more noise than in the second one. Then, as before, we connect them by a geodesic, compute the distance in between them, and also compute some intermediate shapes along the geodesic. The normalized distance between them is relatively small. It is $d = 0.09$. So, we observe that the distance on $\mathcal{GS}_{\mathrm{n,p}}$ is resistant to additive noise in the location of feature points.

6.5 Application: Data-driven sub-manifolds on $\mathcal{GS}_{n,p}$

The shape space $\mathcal{GS}_{\mathrm{n,p}}$ is very vast and high-dimensional. It includes shapes that are not likely to exist in the real world. The shortest distance geodesic connecting the two shapes that are probable of occurring is likely to pass through the shapes that are not probable. Thus, the shortest path through probable shapes in the shape space should not be a straight-line geodesic path of $\mathcal{GS}_{\mathrm{n,p}}$.

It has been long recognized in the computer vision community that objects produce views that in the space of all possible views form surfaces that are generally low dimensional and smooth. For example, [30] studies the ability of human subjects to form the low-dimensional shape space. Similarly, [10] studies the problem of image clustering by introducing a method for surface clustering whose objectives are to detect low-dimensional smooth surfaces. Also, in pattern recognition community, the use of word "geodesic" coincides with the problem of learning the low-dimensional smooth surface embedded in the Euclidean space and path following within the surface. This problem is commonly referred to as data-driven manifolds.

One can naturally combine the concepts for data-driven manifolds in the Euclidean space and apply them to the shape space. In a way, given enough data samples that are probable shapes in the high-dimensional shape space $\mathcal{GS}_{\mathrm{n,p}}$ one would like to learn the low-

dimensional submanifold of $\mathcal{GS}_{\mathrm{n,p}}$ of probable shapes. In order to achieve this, one should use the intrinsic similarity measure of $\mathcal{GS}_{\mathrm{n,p}}$ instead of the Euclidean similarity measure that is typically used in the data-driven manifolds. A good reference for data-driven manifolds in the Euclidean space is given by [87]. Data-driven low-dimensional submanifolds are generally represented by graphs. These concepts are also closely related to level sets and regularizations, see [106, 50].

Related to these concepts are an extension of Principal Component Analysis (PCA) to the Riemannian manifolds. PCA is a prevalent technique for describing model variability. However, it is only applicable when model parameters are elements of a Euclidean vector space. One defines Gaussian distributions on the Riemannian manifolds and derives the maximum likelihood estimates of the mean and covariance. Using these distributions, then one introduces Principal Geodesic Analysis (PGA), the extension of PCA to Riemannian manifolds, see [36]. This is done by first finding the sample mean, and then performing the rest of statistical calculations in the tangent space, attached at the mean.

6.6 Summary

In this chapter we started by presenting a real world application — multi-author handwritten digit recognition. Based on the experimental results of this application, we showed that the metric we chose in the shape space does achieve good results in determining similarity of two shapes. Moreover, we showed that when the linear approximation to the barycentric combinations is used in practice, errors should be negligible for most real world applications. Related to it, we also discussed the geodesic triangles on the Grassmann manifold $\mathcal{G}r_{\mathrm{n,p}}$. In addition, we showed additional experimental results for shape morphing in the shape space $\mathcal{GS}_{\mathrm{n,p}}$, when we take two shapes and connect them by a minimal geodesic segment. Finally, we discussed a need for data-driven low-dimensional submanifolds of the shape space, that contain the probable shapes.

7. CONCLUSIONS

The thesis addresses the problems of computations on the Grassmann manifold and affine-permutation invariant shape space representation as a locally Riemmanian manifold. In our work, we proceed from the fact that the similarity measure is not an Euclidean metric. It is shown that there is no easy injective map between the set of all shapes and some Euclidean space. We adopt the concept of a shape given by Kendall. However, our research is different in a number of ways. If Kendall assumes a labeled finite subset of object's feature points, in our case, the feature points are not labeled, causing their permutation distortions. Also, the class of allowed motions in Kendall's work and in our work is quite different. Kendall restricts the class of motions to rigid motions and uniform scaling. In our research, we are modeling a class of affine motions. Moreover, permutation distortions of the unlabeled feature points pose computationally infeasible combinatorial problem. Combined together with the affine distortions, it is even more challenging problem. In order to simultaneously solve the correspondence problem and the affine invariance problem reliably and minimize the combined affine-permutation shape distortion, this thesis makes a number of substantive original developments.

A novel approach to measuring similarity between shapes and finding the geometric manifold with an unlabeled-view-independent metric property defined on it is established, and key concepts on the representation of geometric objects under different viewing conditions are developed. One of the essential concepts here is defining a shape to be a set of all equivalent representations obtained by acting on a rigid object with a class of distortions and then mapping each shape as a point of a geometric manifold. The underlying metric of the non-Euclidean space intrinsically defines an unlabeled-view-independent similarity measure

between the shapes. We consider affine distortions and distortions induced by permutation of the unlabeled feature points. We solve this challenging problem of finding a similarity measure and of finding a locally geometric manifold with an unlabeled-view-independent metric properly defined on it. We construct geometric shape spaces for 2D and 3D rigid objects and define on these spaces distances among shapes that are invariant to a class of distortions — affine transformations and permutations of unlabeled feature points.

At the first step we consider affine-invariant shape space representation by points on the real Grassmann manifold. At this step we choose to stay within the space of orbits because an operation of finding a canonical member is not an easy one, and is noise sensitive. A typical approach that is taken by in the literature for this step is defining principal axes of the shape and then un-rotating the shape to align its principal axes with their canonical position. However, even small noise can introduce a large change in a position of principal axes causing discontinuities in the similarity measure. We avoid this approach by staying within the space of orbits and developing formulas on the Grassmann manifold that are invariant to the choice of the representative of each equivalency class. We introduce an injection that maps shapes into points of a Riemannian manifold. Since shapes are represented by points on a manifold, algorithms for the Euclidean space can no longer be used, and equivalent algorithms for the Riemannian space are created. We introduce the logarithmic map, which we presented for the first time at the conference in St. Luis in Sep 2003, [98]. We relate the logarithmic map — the inverse of a well-known exponential map, with the concepts of expected values and barycentric combinations. Then we focus on efficient geometric algorithms on the Grassmann manifold. We introduce the numerical representation of points on the Grassmann manifold and derive original formulas that allow computation of basic building blocks required for most algorithms, such as, distance between two points, barycentric combination of k points, sample mean and higher order statistics. Instead of defining an equation for the geodesic velocity vector field, given an initial point and a tangent direction at the point, we define this equation, given an initial point and a final point on the Grassmann manifold, which are to be connected by a geodesic segment. A

closed form linear solution for the midpoint of two points, for the geodesic segment and its derivative vector field is obtained. Also, a linear algorithm for the barycentric combination of k points is developed. The derived original general-purpose explicit formulas for the real Grassmann manifold allow easy computation of similarity in between the shapes as well as a sample mean and higher order moments of a constellation of shapes. The development of a new modeling and computational framework, and related general-purpose calculation formulas and algorithms we pioneered is one of the main results of the thesis.

At the second step we consider a more general case of affine-permutation invariance in shape representation. We develop an analytical mapping to the neighborhood of the real Grassmann manifold as a solution to the distance minimization problem. We define equivalency sets of configurations and describe an abstract representation of these equivalency classes as points in a high-dimensional Riemannian manifold. Each equivalency class collects the configurations of p-dim objects that are transformable into each other under a class of affine transformations and a class of permutations of the feature points. This completely removes a source of affine-permutation shape distortion. We consider both 3D and 2D shapes of rigid objects and construct geometric shape spaces with distances defined in such a way that they achieve invariance to a class of modeled distortions, in particular, simultaneously applied affine transformations and permutations of unlabeled feature points. As in the affine-permutation invariant shape space the distance is not easily computable, we relax an optimization problem to doubly stochastic matrices and present an approximate solution using linear programming and quadratic programming. It is easy to compute and is accurate if the shapes are not too far apart.

Finally, we present applications of our innovative approach, basic concepts and derived formulas to the classification problem on shape spaces. We show how to compute sample mean for a constellation of shapes. We give an example of shape morphing by means of connecting two shapes with a minimal geodesic segment and following it. We also give an example of shape identification when applied to multi-author handwritten digit recognition. We also discuss data-driven sub-manifolds of the shape space. The obtained results demon-

strate the practical importance of the findings and can be used for various computer vision and image processing case studies.

The findings presented in the thesis bridge a gap between image processing algorithms in Riemannian spaces and conventional algorithms embedded for simplicity in Euclidean spaces, and allow to view these algorithms from a new perspective.

Future directions

The majority of available numerical techniques for evaluation and optimization assume an underlying Euclidean space. Yet many computational problems can be properly solved only in non-Euclidean spaces. There were attempts to develop abstract algorithms that exploit the underlying geometry of Riemannian manifolds, but the conversion of these abstract geometric algorithms into numerical procedures for real life applications has proved to be challenging. Future work will be conducted having in mind to overcome the limitations of the existing methodology and extend the application area of the non-Euclidean spaces based on derived closed-form formulas. Many such examples can be given beyond the shape representation theory considered in the thesis.

Statistical inference on the Riemannian manifolds is a rather new subject and can be very essential for the related sciences. It will deepen and broaden the research area to many fields, such as astronomy, meteorology, geological sciences, biology, medicine, physics, chemistry, computer science, signal processing, communication systems, control systems, econometrics, computational finance, etc.

We will focus our efforts on the field of computational finance, where the obtained results, formulas and algorithms can provide a unique framework for the study of many real problems of financial markets and their interactions, particularly of the closely linked problems in asset pricing, risk management and portfolio optimization.

LIST OF REFERENCES

[1] P. A. Absil, R. Mahony, and R. Sepulchre. Riemannian geometry of Grassmann manifolds with a view on algorithmic computations. *Acta Applicandae Mathematicae*, 80(2):199–220, Jan 2004. 60, 62, 74

[2] P. M. Q. Aguiar and J. M. F. Moura. Factorization as a rank 1 problem. *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 178–184, 1999. 13

[3] F. Alimoğlu and E. Alpaydin. Methods of combining multiple classifiers based on different representations for pen-based handwriting recognition. *Proc. of the Fifth Turkish AI and Artificial Neural Networks Symposium, Istanbul, Turkey*, 1996. 102, 106

[4] F. Alimoğlu and E. Alpaydin. Combining multiple classifiers for pen-based handwritten digit recognition. *Elektrik: Turkish J. of Elec. Eng. and Comp. Sc.*, 9(1):1–12, 2001. 102, 106

[5] S. I. Amari, O.E. Barndorff-Nielsen, R. E. Kass, S. L. Lauritzen, and C. R. Rao. *Differential Geometry in Statistical Inference*, volume 10 of *Lecture Notes — Monograph Series*. Institute of Mathematical Statistics, Hayward, CA, 1987. 47

[6] R. V. Ambartzumian. *Factorization, Calculus and Geometric Probability*. Cambridge University Press, 1990. 8

[7] M. A. Armstrong. *Groups and Symmetry*. Springer-Verlag, 1988. 34, 47

[8] A. Baker. *Matrix Groups: An Introduction to Lie Group Theory*. Springer-Verlag London, 2003. 28, 34, 47, 52

[9] S. Baker and I. Matthews. Equivalence and efficiency of image alignment algorithms. *IEEE Conf. on Computer Vision and Pattern Recognition*, 1:1090, 2001. 6

[10] R. Basri, D. Roth, and D. Jacobs. Clustering appearances of 3D objects. *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 414–420, Jun 1998. 115

[11] E. Begelfor and M. Werman. Affine invariance revisited. *IEEE Conf. on Computer Vision and Pattern Recognition*, (to be published), Jun 2006. 9, 28, 29

[12] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. Patt. Anal. Mach. Intell.*, 24(24), 2002. 6

[13] M. Berger. *A Panoramic View of Riemannian Geometry*. Springer, Berlin, 2003. 34, 49

[14] R. Berthilsson. A statistical theory of shape. *SSPR/SPR*, pages 677–686, 1998. 8

[15] A. Bjorck and G. H. Golub. Numerical methods for computing angles between linear subspaces. *Math. Comp.*, 123:579–594, 1973. 66

[16] W. M. Boothby. *An Introduction to Differentiable Manifolds and Riemannian Geometry*. Academic Press, 1975. 27, 28, 29, 34, 65, 83

[17] A. Borovik and A. Nesin. *Groups of finite Morley rank*. Oxford Univ. Press, 1994. 16

[18] R. W. Brocket. Singular values and least square matching. *Proc. on Decision and Control*, (36):1121–1124, Dec 1997. 89

[19] R. W. Brockett. Least squaers matching problems. *Linear Algebra and its Applications*, 122/123/124:761–777, 1989. 89

[20] R. W. Brockett. Dynamical systems that sort lists and solve linear programming problems. *Proc. on Decision and Control*, 146(27):79–91, 1991. 89

[21] M. D. Carmo. *Riemannian Geometry*. Birkhăuser, Boston, 1992. 34

[22] J. Chao and S. Ishii. Invariant recognition and segmentation of 3D object using Lie algebra models. *ICIP Proc. on Image Processing*, 1:550–554, 1999. 8

[23] I. Chavel. *Riemannian Geometry: A Modern Introduction*. Cambridge University Press, 1993. 34

[24] M. T. Chu. Scaled Toda-like flows. *Linear Alg. Appl.*, 215:261–273, 1995. 89

[25] J. N. Clelland. *Lie Groups and the Method of Moving Frames*. MSRI Workshop, July 1999. 14

[26] K. De Cock and B. De Moor. A conjecture on Lyapunov equations and principal angles in subspace identification. *Unsolved Problems in Mathematical Systems and Control Theory*, Princeton University Press:287–292, 2004. 66

[27] J. H. Conway, R. H. Hardin, and N. A. Sloane. Packing lines, places, etc: packings in Grassmannian spaces. *Experimental Mathematics*, 5(2):139–159, 1996. 57

[28] W. J. Culver. On the existence and uniqueness of the real logarithm of a matrix. *Proc. Amer. Math. Soc.*, 17:1146–1151, 1966. 52

[29] M. L. Curtis. *Matrix Groups*. Springer-Verlag, New York, Heidelberg, 1979. 34

[30] F. Cutzu and S. Edelman. Explorations of shape space. *CS-TR 95-01, Weizmann Institute of Science*, 1995. 115

[31] Z. Drmac. On principal angles between subspaces of Euclidean space. *SIAM J. Matrix Anal. Appl.*, 22(1):173–194, 2000. 66

[32] I. L. Dryden and K. V. Mardia. *Statistical Shape Analysis*. John Wiley and Sons, 1998. 8

[33] A. Edelman, T. A. Arias, and S. T. Smith. The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix Anal. Appl.*, 20(2):303–353, 1998. 50, 57, 60, 62, 63

[34] R. Ferreira and J. Xavier. Hessian of the Riemannian squared distance function on connected locally symmetric spaces with applications. *Controlo 2006 — 7th Portuguese Conf. on Automatic Control*, Sep 2006. Instituto Superior Técnico, Lisbon, Portugal. 49

[35] R. Ferreira, J. Xavier, J. P. Costeira, and V. Barroso. Newton method for Riemannian centroid computation in naturally reductive homogeneous spaces. *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, (2504), May 2006. 49

[36] P. T. Fletcher, S. Joshi, C. Lu, and S. Pizer. Gaussian distributions on Lie groups and their application to statistical shape analysis. *Inf. Proc. Medical Imaging*, 2732(18):450–462, 2003. 116

[37] D. A. Forsyth and J. Ponce. *Computer Vision: A Modern Approach*. Prentice Hall, 2003. 5, 13

[38] J. Gallier. *Geometric Methods and Applications*. Springer-Verlag New York, Inc., 2001. 14, 34, 50, 52, 53, 57, 60

[39] J. Gallier and D. Xu. Computing exponentials of skew-symmetric matrices and logarithms of orthogonal matrices. *International J. of Robotics and Automation*, 17(4):1–11, 2002. 47, 52

[40] G. H. Golub and C. F. Van Loan. *Matrix Computations*. The John Hopkins University Press, 3rd edition, 1996. 66

[41] V. V. Gorbatsevich, A. L. Onishchik, and E. B. Vinberg. *Foundations of Lie Theory and Lie Transformation Groups*. Springer-Verlag, 1997. 34

[42] A. Graham. *Kronecker Products and Matrix Calculus with Applications*. Ellis Horwood Ltd., a Division of John Wiley and Sons, 1981. 94, 95, 96

[43] U. Grenander and D. M. Keenan. On the shape of plane images. *SIAM Journal of Applied Mathematics*, 53(4):1072–1094, 1993. 8

[44] U. Grenander, M. Miller, and A. Srivastava. Hilbert-Schmidt lower bounds for estimators on matrix Lie groups for ATR. *IEEE Trans. Patt. Anal. Machine Intell.*, 20:790–802, 1998. 8

[45] V. H. S. Ha. *Blind Recovery of Intrinsic Shape*. PhD thesis, Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA, August 2002. 7, 9, 26, 31, 60

[46] V. H. S. Ha and J. M. F. Moura. Intrinsic shape. *36th Asilomar Conference on Signals, Systems and Computers*, 2:993–997, Nov 2002. 9

[47] V. H. S. Ha and J. M. F. Moura. Efficient 2D shape orientation. *International Conference on IEEE Image Processing*, 1:225 – 228, Sep 2003. 9

[48] V. H. S. Ha and J. M. F. Moura. Three-dimensional intrinsic shape. *International Conference on Image Processing*, pages 2099–2102, 2004. 9

[49] B. C. Hall. *Lie Groups, Lie Algebras, and Representations*. Springer-Verlag, 2003. 34

[50] J. Haslinger and R. A. E. Mäkinen. *Introduction to Shape Optimization: Theory, Approximation and Computation*. Soc. for Indistrial and Applied Math., 2003. 116

[51] C. He and J. M. F. Moura. Robust detection with the gap metric. *IEEE Transactions on Signal Processing*, 45(6):1591–1604, 1997. 57

[52] S. Helgason. *Differential Geometry, Lie Grous and Symmetric Spaces*. Academic, New York, 1978. 45, 60, 65

[53] N. J. Higham. Evaluating Pade approximants of the matrix logarithm. *SIAM J. Matrix Anal. Appl.*, 22(4):1126–1135, 2001. 52

[54] S. Hildebrandt. Harmonic mappings of Riemannian manifolds. *In: Giusti, E. ed., Harmonic Mappings and Minimal Immersions. Lec. Notes in Math.*, 1161:1–117, 1985. 48, 49

[55] W. V. D. Hodge and D. Pedoe. *Methods of Algebraic Geometry*. Cambridge, England: Cambridge University Press, 1952. 57

[56] R. Horn and C. Johnson. *Matrix Analysis*. Cambridge University Press, 1985. 92

[57] K. Hüper, J. Götze, and S. Paul. Subspace separation by discretizations of double bracket flows. *(unpublsihed)*, http://citeseer.ifi.unizh.ch/180616.html, 2000. 89

[58] D. P. Huttenlocher and S. Ullman. Recognizing solid objects by alignment with an image. *Int. J. Comp. Vision*, 5(2):195–212, 1990. 6, 14

[59] C. Jordan. Essai sur la géométrie à n dimensions. *Bulletin de la Société Mathématique*, 3:103–174, 1875. 66

[60] S. C. Joshi, M. I. Miller, and U. Grenander. On the geometry and shape of brain submanifolds. *International Journal of Pattern Recognition and Artificial Intelligence*, 11:1317–1343, 1997. 8

[61] J. Jost. *Riemannian Geometry and Geometric Analysis*. Springer-Verlag, Berlin, 2002. 44

[62] H. Karcher. Riemannian center of mass and mollifier smoothing. *Comm. Pure Appl. Math.*, 30:509–541, 1977. 48

[63] T. Kato. *Perturbation Theory for Linear Operators*. New York: Springer-Verlag, 2nd edition, 1976. 66

[64] D. G. Kendall. The diffusion of shape. *Advances in Appl. Prob.*, 9:428–430, 1977. 7

[65] D. G. Kendall, D. Barden, T. K. Carne, and H. Le. *Shape and Shape Theory*. John Wiley and Sons, 1999. 1, 8, 47, 48, 49

[66] W. S. Kendall. Probability, convexity and harmonic maps with small image I: uniqueness and fine existence. *Proc. London Math. Soc.*, 61:371–406, 1990. 48, 49

[67] A. V. Knyazev and M. E. Argentati. Principal angles between subspaces in an A-based scalar product: algorithms and perturbation estimates. *Society for Industrial and Applied Mathematics*, 23(6):2009–2041, 2002. 66

[68] S. Kobayashi and K. Nomizu. *Foundations of Differential Geometry*, volume 1-2. John Wiley and Sons, 1963. 34, 38

[69] T. Kohonen. *Self-organizing maps*. Springer Series in Information Sciences, 30. Springer-Verlag, Berlin, 3rd edition, 2001. 48

[70] J. M. Lee. *Introduction to Smooth Manifolds*. Springer-Verlag, 2000. 34, 37

[71] Von K. Leichtweiss. Zur Riemannschen geometrie in Grassmannschen mannigfaltigkeiten. *Math. Zeitschr.*, 76:334–366, 1961. 57

[72] J. Maciel. *Global Matching: Optimal Solution to Correspondence Problems*. PhD thesis, Instituto Superior Técnico,Universidade Técnica de Lisboa, Lisbon, Portugal, August 2001. 6, 23, 89, 95

[73] J. Maciel and J. P. Costeira. A global solution to sparse correspondence problems. *IEEE Trans. Patt. Anal. Mach. Intell.*, 25(2):187–199, 2003. 6, 23, 89, 95

[74] C. S. Macinnes. The solution to a structured matrix approximation problem using Grassman coordinates. *SIAM J. Matrix Anal. Appl.*, 21(2):446–453, 1999. 57

[75] S. MacLane and G. Birkoff. *Algebra*. The MacMillan Company, 1967. 66, 92

[76] J. R. Magnus and H. Neudecher. *Matrix Differential Calculus*. John Wiley and Sons, 1988. 62, 94, 96

[77] M. I. Miller, A. Trouvé, and L. Younes. Geodesic shooting for computational anatomy. *J. of Math. Imaging and Vision*, 24(2):209 – 228, Mar 2006. 8

[78] M. Moakher. Means and averaging in the group of rotations. *SIAM J. Matrix Anal. Appl.*, 24(1):1–16, 2002. 53

[79] C. B. Moler and C. F. Van Loan. Nineteen dubious ways to compute the exponential of a matrix. *SIAM Review*, 20:801–836, 1979. 52

[80] J. B. Moore, R. E Mahony, and U. Helmke. Numerical gradient algorithms for eigenvalue and singular value calculations. *SIAM J. Matrix Anal. Appl.*, 15(3):881–902, 1994. 89

[81] D. Mumford. Mathematical theories of shape: Do they model perception? *Proc. SPIE, Geometric Methods in Computer Vision*, 1570:2–10, 1991. 101

[82] J. L. Mundy and A. Zisserman, editors. *Geometric Invariance in Computer Vision*. The MIT Press, Cambridge, MA. 6

[83] P. M. Murphy and D. W. Aha. UCI Repository of machine learning databases. http://www.ics.uci.edu/˜mlearn/MLRepository.html(1994). Irvine, CA: University of California, Department of Information and Computer Science. 102

[84] M. K. Murray and J. W. Rice. *Differential Geometry and Statistics*, volume 48 of *Monographs on Statistics and Applied Probability*. Chapman and Hall / CRC, 1993. 47

[85] K. Nagao and W. E. L. Grimson. Affine matching of planar sets. *Computer Vision and Image Understanding*, 70(1):1–22, 1998. 6, 14

[86] R. K. Nagle, E. B. Saff, and A. D. Snider. *Fundamentals of Differential Equations and Boundary Value Problems*. 3rd edition, 1999. 36, 74

[87] P. Niyogi, F. Girosi, and T. Poggio. Incorporating prior information in machine learning by creating virtual examples. *Proc. IEEE*, 86(11):2196–2209, 1998. 116

[88] D. Nowicki and O. Dekhtyarenko. Averaging on Riemannian manifolds and unsupervised learning using neural associative memory. *Proc. European Symposium on Artificial Neural Networks*, 48:181–186, 2005. 57, 77

[89] J. M. Oller and J. M. Corcuera. Intrinsic analysis of statistical estimation. *The Annals of Statistics*, 23(5):1562–1581, Oct 1995. 45, 48, 49

[90] C. C. Paige and M. Wei. History and generality of the CS decomposition. *Linear Algebra Appl.*, 208/209:303–326, 1994. 66

[91] X. Pennec. Uniform distribution, distance and expectation problems for geometric feature processing. *J. of Math Imaging and Vision*, 9(1):49–67, Jul 1998. 49

[92] X. Pennec. Probabilities and statistics on Riemannian manifolds: Basic tools for geometric measurements. *Proc. of Nonlinear Signal and Image Processing*, 1:194–198, 1999. 49

[93] X. Pennec. Probabilities and statistics on Riemannian manifolds: A geometric approach. *Submitted to Int. J. of Math. Imaging and Vision*, Jan 2004. 49

[94] P. O. Persson and G. Strang. A simple mesh generator in MATLAB. *SIAM Reviews*, 46(2):329–345, Jun 2004. 101

[95] T. Sakai. On cut loci on compact symmetric spaces. *Hokkaido Math J.*, 6:136, 1977. 66

[96] T. Sakai. *Riemannian Geometry*, volume 149. AMS, 1996. 45, 60

[97] G. L. Scott and H. C. Longuet-Higgins. An algorithm for associating the features of two images. *Proc. R. Soc. Lond. B*, 244:21–26, 1991. 6, 88

[98] D. A. Sepiashvili, J. M. F. Moura, and V. H. S. Ha. Affine-permutation symmetry: Invariance and shape space. *Statistical Signal Processing, 2003 IEEE Workshop*, pages 307– 310, Sep 2003. 9, 29, 74, 118

[99] J. Shi and C. Tomasi. Good features to track. *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 593–600, Jun 1994. 88

[100] S. T. Smith. *Geometric Optimization Methods for Adaptive Filtering*. PhD thesis, Harvard University, Cambridge, Massachusetts, 1993. 5

[101] S. T. Smith. Optimization techniques on Riemannian manifolds, in Hamiltonian and gradient flows, algorithms and control. *A. Bloch, ed., AMS, Providence, RI*, pages 113–136, 1994. 53

[102] S. T. Smith. Covariance, subspace, and intrinsic Cramér-Rao bounds. *IEEE Trans. on Sig. Proc.*, 53(5):1610–1630, May 2005. 5, 69, 78

[103] M. Spivak. *A Comprehensive Introduction to Differential Geometry*, volume 1-2. Houston, TX: Publish or Perish, Inc., 3rd edition, 1999. 34

[104] A. Srivastava and D. R. Fuhrmann. Gradient flows on projection matrices for subspace estimation. *Proc. Asilomar Conf.*, (31), 1997. 89

[105] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. *International J.of Computer Vision*, 9(2):137–154, 1992. 13

[106] R. M. Willett and R. D. Nowak. Minimax optimal level set estimation. *submitted to IEEE Transactions on Image Processing*, Feb 2006. 116

[107] Y. C. Wong. Differential geometry of Grassmann manifolds. *Proc. Nat. Acad. Sci. USA*, 57:589–594, 1967. 66, 67

[108] Y. C. Wong. Conjugate loci in Grassmann manifolds. *Bull. Am. Math. Soc.*, 74:240, 1968. 66

[109] Y. C. Wong. Sectional curvatures of Grassmann manifolds. *Proc. Natl. Acad. Sci.*, 60:75–79, 1968. 69

[110] J. Xavier. *Blind Identification of MIMO Channels Based on 2nd Order Statistics and Colored Inputs*. PhD thesis, Instituto Superior Técnico, Lisbon, Portugal., 2002. 5, 27, 60

[111] A. Yezzi and A. C. G. Mennucci. Metrics in the space of curves. *arXiv (unpublished)*, http://www.citebase.org/cgi-bin/citations?id=oai:arXiv.org:math/0412454, 2004. 8

[112] C. Zanella. Embeddings of Grassmann spaces. *J. of Geometry*, 52:193–201, Jan 1995. 57

[113] L. Zheng and D. N. C. Tse. Communication on the Grassmann manifold: A geometric approach to the noncoherent multiple-antenna channel. *IEEE Trans. Informat. Th.*, 48(2):359–383, 2002. 60