Modelling Shapes with Intermittently Present Landmarks

Gérard Sanromà

Advisors: Francesc Serratosa
René Alquézar
Modelling Shapes with Intermittently Present Landmarks

Gerard Sanromà
Advisors: Francesc Serratosa
René Alquézar

June 4, 2009
The analysis of shape is of great importance in a wide variety of applications. Point Distribution Models (PDM) describe how to create shape models which allow for considerable variability but are still specific to the class of structures they represent. Nevertheless, it is needed the shapes inside the class to have the same amount of landmark points in order to be modelled with PDM. However, there are applications where objects present intermittence in some of their features, thus leading to shapes with different number of landmark points. We present in this document a new compact model to represent a data set based on the main idea of the Point Distribution Model (PDM). Our model overcomes PDM in two aspects, first, it is not needed all the objects to have the same number of points. This is a very important feature, since in real applications, some objects may present intermittence in some of their landmarks. And second, the model captures the non-linearity of the data set. Some research have been presented that couples both aspects separately but no model have been presented until now that couples them as a whole. A case study shows the efficiency of our model and its improvement respect the models in the literature.
Contents

1 Introduction 3
  1.1 Introduction ........................................... 3
  1.2 Our Contributions ..................................... 5
  1.3 Organization .......................................... 5

2 State-of-the-art in Statistical Shape Analysis 6

3 Statistical Shape Analysis 13
  3.1 Introduction ........................................... 13
  3.2 Configuration Space and Size Measures ................... 14
    3.2.1 Configuration Space .............................. 15
    3.2.2 Size ............................................. 15
  3.3 Shape Space ........................................... 16
    3.3.1 Introduction ..................................... 16
    3.3.2 Filtering translation ............................. 16
    3.3.3 Pre-shape ....................................... 17
    3.3.4 Shape ........................................... 17
    3.3.5 Size-and-shape: Removing location and rotation ..... 18
  3.4 Distances ............................................... 19
    3.4.1 Procrustes distances ............................. 19
    3.4.2 Alternative distances ............................ 21
    3.4.3 Tangent space coordinates ....................... 23
  3.5 General Procrustes Methods ........................... 25
    3.5.1 Introduction ..................................... 25
    3.5.2 Full ordinary Procrustes analysis ................. 25
    3.5.3 Generalized Procrustes Analysis ................. 27
    3.5.4 Partial Procrustes Analysis ..................... 31

4 Modelling Shape Variability 35
  4.1 Principal Components Analysis .......................... 35
  4.2 Linear Shape Modelling ................................. 38
    4.2.1 Point Distribution Models ....................... 38
  4.3 Non-Linear Shape Modelling ............................ 41
    4.3.1 Polynomial Regression Point Distribution Models ..... 45
5 Modelling Shape Variability with Intermittently Present Landmarks

5.1 Imputation Methods ........................................... 50
  5.1.1 Mean Imputation ......................................... 50
  5.1.2 PCA Imputation .......................................... 50
  5.1.3 PR-PCA Imputation ........................................ 52

5.2 Structured Point Distribution Models .................................. 53

5.3 Polynomial Regression Structured Point Distribution Models .... 55

5.4 Experiments and Results ........................................... 57
  5.4.1 PDM and PRPDM .......................................... 57
  5.4.2 SPDM and PR-SPDM ....................................... 62

5.5 Discussion .......................................................... 62
  5.5.1 Data Imputation .......................................... 62
  5.5.2 Shape modelling ............................................ 67

5.6 Conclusions and Future Work ....................................... 67

6 Publications derived from this Research Project .................... 68
Chapter 1

Introduction

1.1 Introduction

Deformable shape models are aimed at modelling the deformations presented by a particular class of objects. The main advantage of these models is that they allow for considerable variability being still specific to the class of structures they represent.

A classical approach to deformable shape modelling is Point Distribution Models [10]. They are based on the Statistical Shape Analysis approach proposed by I. Dryden and K. Mardia [16].

In all the document, we focus on the case where the objects are represented by a set of key points called landmarks.

These models are often applied to image segmentation [6] and also a number of pattern recognition problems.

Nevertheless, little effort has been done in modelling objects with intermittently present landmarks [37].

In order to model the variability of a shape by the means of classical methods, there must exist a point-wise correspondence among the shapes. This is not the case of many objects that may present, in some of their components, arbitrary patterns of presence or absence. In case of classical methods to deformable shape modelling, one would use only the landmarks which are always present, discarding those which may appear or disappear; or may use different models grouping the shape instances with the same number of landmarks. These options are not recommendable in many applications because patterns of presence or absence are also a source of information. Some examples where intermittently present landmarks modelling would be useful are:

1. Diagnosis based on medical image analysis such as diabetic nerve capillaries [37]. See Figure 1.1. In disease condition, such as diabetic neuropathy, changes occur including constriction of the lumen. In some cases the lumen can become so constricted as to be unidentifiable 1.1(b).
Figure 1.1: Microscope images of nerve capillaries. There are several concentric layers of structures in capillary cross-sections. The central region is the lumen: the space through which blood cells pass 1.1(a). Extracted from [37]

Figure 1.2: MRI of horizontal cross sections of a human brain. One can observe complex patterns of apparition/disapparition of different structures, e.g., the superior colliculus, a structure nearly at the center of the horizontal cross-sections that can only be observed in the cross-sections of the lower parts of the brain

2. Study of histological cross-sections, in which structures may appear in a proportion of contiguous slices in a stack. e.g. horizontal cross-sections of a human brain. See Figure 1.2.

3. Modelling of objects that can or cannot present some features: e.g. faces with or without moustache.

4. Study those type of variations in an object that can produce occlusions, e.g. 3D rotations of the head. See Figure 1.3.

What all the applications have in common is that we want to incorporate the presence or absence of the features as one characteristic more of the variation.

The only approach that permits shapes with intermittently present landmarks is Structured Point Distribution Models (see Section 5.2). This is an approach by Rogers and Graham [37]. It is based on the classical PDM and combines data imputation with a model of structural variation. Data imputation is the pre-processing step where we fill the missing data with some values. As we will see later, the method used will affect to the resulting model.
Structured Point Distribution Models (SPDM) has proven to be successful in capturing the patterns of presence and absence of the landmarks, however there are some situations where it fails to capture shape variability due to the non-linear nature of the variation.

1.2 Our Contributions

With the aim of extending the intermittently present data modelling to the non-linear case, we have defined a new model called Polynomial-Regression Structured Point Distribution Models (see Section 5.3). This new model is thought to be used in combination with a novel imputation scheme also developed by us, called PR-PCA (Section 5.1.3). The aim is to impute the missing data according to their non-linear relationships. We obtain good results in a synthetic example (see results in section 5.4.2).

1.3 Organization

This work is organized as follows: in Chapter 2 we present the state of the art in shape analysis. In Chapter 3 we present methods for measuring, describing and comparing shapes. We also present the shape space and propose various alternatives of shape distances in the shape space. We also describe methods for obtaining an average shape from a population. In Chapter 4 we describe the method of computation of the modes of variation of shape. Both linear and non-linear case are discussed. Finally, in Chapter 5 we present the techniques for building models of shapes with intermittently present landmarks. We study both the linear and non-linear cases.
Chapter 2

State-of-the-art in Statistical Shape Analysis

Statistical shape analysis involves methods for describing, measuring and comparing shapes. A good survey can be found in [25]. These methods often involve the matching of two or more shapes in order to either compute the distance between two shapes or to build a statistical model of shape variation. The methods described in our work assume the shapes summarized by a set of landmarks points, and there must exist a pairwise correspondence between these points. Moreover, in general it is not needed the shapes to be represented by a set of pairwise correspondent landmark points in order to match them. In the following we present various alternatives to deal with shape matching that do not necessarily need the shapes to be compound by a set of pairwise correspondent landmark points.

Serge Belongie et al. [1] consider matching shapes using the context information. First the shapes are discretized by equally spaced points, and then a descriptor called shape context is attached to each point. This descriptor consists on a bidimensional histogram informing on the number of points placed in a certain distance in a certain angle from a given one; thus capturing the distribution of the remaining points relative to a reference point. Corresponding points on two similar shapes may have similar shape contexts. Given the correspondences, the transformation that best aligns the shapes is computed as an optimization problem. Regularized thin-plate splines provide a flexible class of transformation maps for this purpose. The dissimilarity between the two shapes is computed as a sum of matching errors between corresponding points.

Another approach for shape matching is by Xiaolei Huang et al. [27]. They use the distance transform image as shape representation instead of a parametric representation. Then, matching is done in two steps: a global one that finds the similarity transformation that best align the shapes using the Mutual Information criterion; and a local one that totally matches the shapes by B-spline-based Incremental Free Form Deformations (IFFD). After these steps, dense one-to-


one correspondences are established. They demonstrate its robustness to noise, severe occlusion, and missing parts. They have also proven its applicability to statistical modelling of anatomical structures, and 3D face scan registration and expression tracking with success.

C. Davatzikos [11] present a method for matching shapes in which point correspondences are established between a particular shape and a template, based on the geometric characteristics of the boundaries. These correspondences are then used to determine the shape transformation which is based on an elastic adaptation of the template.

Andrew Hill et al. [26] define a framework for automatic landmark identification on a a population of shapes. It consists on building a binary tree where each pair of leaves correspond to a shape and its best matching pair among the remaining population. The correspondences between these pair of shapes are found by matching a pair of sparse polygonal approximations by minimizing a cost function. Then the mean shapes between each pair are calculated and the algorithm is repeated taking as these mean shapes as the new population of shapes. This procedure is repeated until only one mean shape remains. Then, Critical Point Detection algorithm is used to choose the points which best represents the mean shape and they are projected back along the branches of the tree towards the leaves. Results are presented for three classes of shapes which exhibit various type of nonrigid deformations.

Nicolae Duta et al. [17] present a method to automatically construct 2D shape models. The modelling system is given a set of training example shapes defined by the coordinates of their contour points. The shapes are automatically aligned using Procrustes analysis and clustered to obtain cluster prototypes. Those shapes considered to be outliers are discarded. Then a new method to obtain point correspondences based on polygonal approximation is used to obtain the correspondences between the shapes in a cluster. This method is invariant to object pose/scale since a priori Procrustes analysis is done. The matching method is independent of the initial relative position/scale of the two objects and does not require any manually tuned parameters. They used this design to learn shapes from manually extracted contours in MR brain images. A quantitative analysis demonstrated results that compare very well to those achieved by manual registration.

Rhodri H. Davies et al. [14] and [13] describe a method for automatically building statistical shape models from a training set of example boundaries/surfaces. It is based on posing the correspondence problem as one of finding the parameterization for each shape in the training set that build the “best” model. They define “best” as that which minimizes the description length of the training set, arguing that this leads to models with good compactness, specificity and generalization ability. Results are given for several different training sets of two dimensional boundaries, showing that the proposed method constructs better models than other approaches including manual landmarking.

J. Zhang et al. [48] describe a method for object representation and recognition where an object and all its similarity transformed versions are identified with a single point in a high-dimensional manifold called the shape space. Ob-
ject recognition is achieved by measuring the geodesic distance in the shape space. This approach is fully automatic, and the point correspondences are based on the extrema points of the Legendre polynomials. This method has shown promising results in airplane boundary recognition and it is relatively insensitive to noise and occlusions.

As shown in the paper above, the analysis of shapes can be done directly in the shape space instead of a linearized approximation as it is the tangent space.

Anuj Srivastava et al. propose a similar approach for shape clustering and learning in which shapes are represented by closed planar curves in $\mathbb{R}^2$ parameterized by the arc length. At this point, it is obvious that the representation method proposed, impose the restriction at the shapes to be formed by a single closed curve, thus mainly treating with silhouettes. They prove that this type of representation leads to a shape space whose geodesics perform more efficiently than the ones obtained by Procrustes analysis using landmarks.

P. Thomas Fletcher et al. [19] present a similar approach using geodesics to build shape models. Their aim is to apply Principal Components Analysis directly in the shape space which is a Riemannian manifold using geodesics instead of linear vectors to extract the main modes, thus they call it Principal Geodesics Analysis. The shape space is derived from the Medial Representation of 3D shapes. The Medial representation used is based on the medial axis of Blum. Results are presented applying this framework in a population of hippocampi in a schizophrenia study.

There exist a wide variety of approaches to shape modelling. In the following we present some of them.

Rasmus Larsen et al. [36] extend the use of PCA in statistical shape analysis to take into account the noise variance. The extensions to PCA are based on adaptation of Maximum Autocorrelation Factor (MAF) which seeks maximum spatial autocorrelation, and Minimum Noise Fraction (MNF) which is a PCA in a metric space defined by a noise covariance matrix estimated from the data. Finally the different subspace methods are compared based on their ability to represent the data.

While classical PDM are constructed by analysis of the positions of a set of landmark points, T. Cootes et al. [7] construct models that analyze continuous warp fields obtained by parametric diffeomorphic warps. These warps are always defined in the same reference frame which allows the parameters controlling the warps be meaningfully compared from one example to another. A linear model is learnt to represent the variations in the warp parameters across the training set. Models can be built either from sets of annotated points or from unlabelled images. In the latter case techniques from non-rigid registration are used to construct the warp fields in order to deform a reference image into each example.

T. F. Cootes and C. J. Taylor [9] propose a method to estimate the density probability function of a data distribution using a mixture of Gaussians. Given a training set of aligned shapes, the use of Kernel-based techniques to represent this distribution are often complex and expensive. They show that a mixture
of gaussians is a much simpler method to generate estimates which allow us to determine plausible and implausible examples of the class. They describe how to calculate it and how it can be used in image search to locate examples of the modelled objects in new images.

T. F. Cootes and C. J. Taylor [8] present a method of combining two approaches to flexible shape modelling. Modal analysis using finite element methods (FEM) generates a set of vibrational modes for a single shape. Point distribution models (PDM) generate statistical modes of shape variation from a set of example shapes. A new approach combining both methods is proposed to build models of shape variation when training sets with only few examples are available. Results for both synthetic and real examples show that the combined version performs better than either the PDM or FEM models alone.

In the same line as the above paper, Christos Davatzikos et al. [12] present an approach to overcome the limitations imposed by a training set with few examples. They use a hierarchical formulation in which the training shapes are segmented in subparts where each part is modelled separately. The idea is to make small enough subparts as to be efficiently modelled with the available examples. Wavelet transform is used for shape representation in each subpart. A hierarchical model is constructed that joins the local models of each subpart with a global model in order to reconstruct the whole shape. Examples in magnetic resonance images of the corpus callosum and hand contours demonstrate that good segmentations in images can be achieved even with as few as five training examples.

Georg Langs et al. [33] noted that shapes with independent moving parts introduced nonlinearities that deteriorated the throughput of the linear models. Although non-linear extensions for shape modelling have been proposed, it is needed a certain amount of user interaction. In a similar way as the paper above, they propose an algorithm to find an optimal subdivision of the shapes into subparts, but with the aim of making each subpart adequate for linear modeling. They use the minimum description length principle to find the optimal subdivision. The proposed method is evaluated on synthetic data, medical images and hand contours.

In the line of nonlinear modelling P.D. Sozou et al. [41] present a non-linear generalization of PDM using Polynomial regression in order to model the nonlinearities presented in some datasets when the landmarks do not vary in a straight line. Their approach is motivated by noting that the eigen-analysis used to extract the modes of variation of a standard PDM can be conceptualized as a sequential process. They prove the applicability of their method in both synthetic and real shape data where the shapes exhibit bendings.

In the same line as the approach above, P.D. Sozou et al. [42] propose a non-linear generalization of PDM using a multi-layer perceptron.

D. Shi et al. [39] present an approach for radical recognition of handwritten chinese characters from images using Kernel PCA and active shape models.

Sami Romdhani et al. [38] present a multi-view shape model with the aim of modelling the highly nonlinear behaviour present in a 2D image when changes in 3D pose of the imaged object. For nonlinear model transformation they use
Kernel PCA based on Support Vector Machines. They prove it to model a face while rotating the head from $-90^\circ$ to $90^\circ$.

Otherwise C.J Twining and C.J Taylor [43] describe a methodology to build models of the variability in a class of shapes using Kernel PCA. They noted that the above method presented by Sami Romdhani et al. was not generally valid and introduce an improvement that behaves correctly. They prove it with synthetic data and real shapes of nematode worms. It is shown that using such model to impose shape constraints during Active Shape Model (ASM) search gives improved segmentations of worm images than those obtained using linear shape constraints.

Active Shape Models (ASM) constitute an application of PDM for image segmentation. Its main algorithm, ASM search, was first described by T. F. Cootes et al. in 1992 [10] and [6]. A wide variety of approaches derived from that work, some of them improving some aspects of it, others proving its applicability to specific problems. In the following we present the originary works of ASM and a variety of derived approaches.

Some of the first papers that described the methodology of ASM were by T. F. Cootes et al. [10] and [6], which present this method for model-based image segmentation. They argue that approaches based on flexible templates sacrifice model specificity in order to accommodate variability, thereby compromising robustness during image interpretation. They use a method for building models by learning patterns of variability from a training set of correctly annotated images, thus leading to a model able to deform only in ways characteristic of the class of the object it represents; and show that they can be used for image search in an iterative refinement algorithm analogous to that employed by Active Contour Models (Snakes) of Kass et al. [30]

Ghassan Hamarneh and Tomas Gustavsson [24] extend 2D active shape models to 2D + time by presenting a method for modeling and segmenting spatio-temporal shapes (ST-shapes). Segmentation results on both synthetic and real data are presented.

Kwok-Wai Wan et al. [45] present an approach for accurate feature extraction on faces under perspective variations. Their approach is based on using separate models for face contour and facial features. Experimental results show an improvement over classical ASM.

Lu Huchuan and Shi Wengang [28] present a method to improve face alignment for the use with active shape models.

Li Yong et al. [47] present an approach that improves the segmentation of the face contour with ASM using specific knowledge of human face.

Wei Wang et al. [46] present an improvement of ASM search by employing the global texture information in a similar way than active appearance models [5]

Berhard Fröba et al. [20] tackle the problem of real-time alignment of active shape models to new object instances at video frame rate.

P.P Smyth et al. [40] describe how they used ASM to locate vertebrae in lateral images of the spine.
Another important field of application of ASM is on facial expression recognition. A good survey can be found in [18].

Changbo Hu et al. [4] propose an approach for recognizing facial expressions on a low dimensional manifold obtained by Isomap embedding. Facial contour features are clustered using a mixture model and then, expression dynamics are first learned and after tracked in the embedded space using ICondensation. Recognition is performed in a cooperative manner, within a common probabilistic framework. The image observation likelihood is derived from a variation of the Active Shape Model algorithm. A specific ASM is learned for each cluster in the manifold, thus improving facial deformation tracking and expression recognition.

Atul Kanaujia and Dimitris Metaxas [29] propose a framework to recognize various expressions by tracking facial features. They use ASM to track feature points in the subspace obtained from localized Non-negative Matrix Factorization. The tracked feature points are used to train conditional model for recognizing prototypic expressions like Anger, Disgust, Fear, Joy, Surprise and Sandness. They use Conditional Random Fields (CRF) to probabilistically predict expressions. Experimental results show that accurately tracked feature shapes provide reliable discriminative cues to robustly recognize facial expressions for an image sequence.

As can be seen, the analysis of shape is of a great importance in a wide variety of applications but usually the shape information is used together with texture information. In the following we present a variety of approaches where both shape and appearance information are used together, leading to robust models for image interpretation.

T. F. Cootes et al. [5] present Active Appearance Models (AAM), a method for interpreting images that contains a statistical model of the shape together with the gray-level appearance of the object of interest. During the training phase there is learnt the relationship between a training image and a synthesised model example. To match to an image the current residuals are measured and the model is used to predict changes to the current parameters. A good overall match is obtained in few iterations, even from poor starting estimates.

A. Lanitis et al. [34] and [35] describe a method for face identification using a flexible model representing shape and gray-level appearance of human faces. The model parameters control both the inter-class and within-class variation. Discriminant analysis techniques are employed to enhance the effect of those parameters affecting inter-class variation, which are useful for classification. Experiments show good reconstruction even with considerable variability in 3D viewpoint, lighting and facial expression.

Thomas Vetter and Nikolaus F. Troje [44] present a model for coding and synthesis of human faces. It is based on the separation of 2D shape and texture information by the use of pixel-by-pixel correspondence among the various images, established through optical flow algorithms. Evaluation is performed by calculating the distance between original images and reconstructed ones.
Volker Blanz and Thomas Vetter [3] present a face recognition system across wide variations in pose and illumination, including cast shadows and specular reflections. This is achieved by estimating the 3D shape and texture of faces from single images. The estimate is achieved by fitting a statistical, morphable model of 3D faces to images. The model is learned from a set of textured 3D scans of heads.
Chapter 3

Statistical Shape Analysis

In a wide variety of disciplines is of great importance to measure, describe and compare the shapes of objects. The field of shape analysis involves methods for the study of the shape of objects where location, rotation and often scale information can be removed. In particular, we focus on the situation where the objects are summarized by key points called landmarks. Statistical shape analysis is concerned with methodology for analysing shape in the presence of randomness. The objects under study could be sampled at random from a population and the main aims of statistical shape analysis are to estimate population average shapes as well as to describe, measure and compare shapes.

In this Chapter we present methods for describing, comparing and measuring shapes of objects, as well as methods for computing an average shape from a population.

The main part of the theory described here has been extracted from the excellent work by Dryden and Mardia [16].

In Section 3.1 we give an introductory definition of the main concepts involved in shape analysis. In Section 3.2 we describe the space where lie the landmark configurations and propose a measure of size. In Section 3.3 we describe the steps involved in extracting a shape from a configuration of landmarks as well as some properties of the space where shapes lie. In Section 3.4 we present various choices of distances between shapes. Finally, in Section 3.5 we present methods for extracting an average shape from a population.

3.1 Introduction

Shape is a property inherent to all objects - natural or man-made. Advances in technology have led to the routine collection of geometrical information and the study of the shapes of objects is increasingly important. Shape analysis is of great interest in a wide variety of disciplines such as biology, medicine, image analysis, archaeology, geography, geology, agriculture and genetics.
The word ‘shape’ is very commonly used in everyday language, usually referring to the appearance of an object. Following D.G. Kendall [31] the definition of shape that we consider is intuitive.

**Shape** is all the geometrical information that remains when location, scale and rotational effects are filtered out from an object.

So, an object’s shape is invariant under the Euclidean similarity transformations of translation, scaling and rotation. For example the shape of a fish consists of all the geometrical properties that are unchanged when the fish is translated, rescaled or rotated.

Sometimes we are also interested in retaining scale information (size) as well as the shape of the object.

**Size-and-shape** is all the geometrical information that remains when location and rotational effects are filtered out from an object.

Two objects have the same size-and-shape if they can be translated and rotated to each other so that they match exactly.

In the last two decades there have been many key developments in shape analysis that allow us to work on the landmark coordinates directly. Also, the advances in technology of measuring landmarks have been helpful, e.g. landmarks from digitized objects. Of course if there were no constraints on the landmarks, then we could use standard multivariate analysis, but in general the statistical methodology for shape is inherently non-Euclidean.

The idea is that, rather than working with quantities derived from organisms, one works with the complete geometrical object itself (up to similarity transformations). The approach is very much in the spirit of D’Arcy Thompson who considered the geometric transformations of one species to another. The important point to note is that he worked with geometrical pictures of organisms rather than derived quantities. Throughout the text it will be observed that pictures of the organisms or objects under study can always be easily constructed and it is this that embodies our geometrical approach to shape analysis. Some of the strengths of the methods that we describe are testing for shape difference and to describe the morphological changes in the objects under study.

In particular, we shall consider a shape space obtained directly from the landmark coordinates, which retains the geometry of a point configuration. Hence, this approach to shape analysis has been called ‘geometric shape analysis’ by various authors.

### 3.2 Configuration Space and Size Measures

In this subsection we define the configuration and configuration space. We also discuss measures of the size of a landmark configuration.
3.2.1 Configuration Space

The configuration is the set of landmarks on a particular object. The configuration matrix $X$ is the $k \times m$ matrix of Cartesian coordinates of the $k$ landmarks in $m$ dimensions. The configuration space is the space of all possible landmark coordinates.

In our applications we have $k \geq 3$ landmarks in $m = 2$ dimensions. Thus, the configuration space is typically $\mathbb{R}^{km}$.

3.2.2 Size

In order to define a shape we should remove the size information. Is for this reason that we must first define what we mean by size. Consider $X$ to be a $k \times m$ matrix of the Cartesian coordinates of $k$ landmarks in $m$ real dimensions, i.e. the configuration matrix of the object.

A size measure $g(X)$ is any positive real valued function of the configuration matrix such that

$$g(aX) = ag(X)$$

for any positive scalar $a$.

The centroid size is given by

$$S(X) = \| CX \| = \sqrt{\sum_{i=1}^{k} \sum_{j=1}^{m} (X_{ij} - \bar{X}_j)^2}, X \in \mathbb{R}^{km},$$

(3.2)

where $X_{ij}$ is the $(i,j)$th entry of $X$, the arithmetic mean of the $j$th dimension is $\bar{X}_j = \frac{1}{k} \sum_{i=1}^{k} X_{ij},$

$$C = I_k - \frac{1}{k} 1_k 1_k^T$$

(3.3)

is the centering matrix, $\| X \| = \sqrt{\text{trace} \left( X^T X \right)}$ is the Euclidean norm, $I_k$ is the $k \times k$ identity matrix, and $1_k$ is the $k \times 1$ vector of ones.

Obviously $S(aX) = aS(X)$ thus satisfying equation 3.1. The centroid size $S(X)$ is the square root of the sum of squared Euclidean distances from each landmark to the centroid, namely

$$S(X) = \sqrt{\sum_{i=1}^{k} \| (X)_i - \bar{X} \|^2},$$

(3.4)

where $(X)_i$ is the $i$th row of $X$ $(i = 1, \ldots, k)$ and $\bar{X} = (\bar{X}_1, \ldots, \bar{X}_m)$ is the centroid. This measure of size will be used throughout the chapter. In fact, the centroid size is the most commonly used size measure in geometrical shape analysis (e.g. Dryden and Mardia [16]).
3.3 Shape Space

In this Section we investigate geometrical aspects of shape. In particular, we describe the pre-shape space and the shape space for any dimensions.

3.3.1 Introduction

We have already noted that in order to obtain the shape of an object we must first remove the geometrical information concerning to the position, rotation angle and size.

A rotation of a configuration is given by post-multiplication of the configuration matrix $X$ by a rotation matrix $\Gamma$.

An $m \times m$ rotation matrix satisfies $\Gamma^T \Gamma = \Gamma \Gamma^T = I_m$ and $|\Gamma| = +1$. The set of all $m \times m$ rotation matrices is known as the special orthogonal group $SO(m)$.

A translation consists in adding a constant $m$-vector to the coordinates of each point. A scaling is obtained by multiplying $X$ by a positive real number.

The Euclidean similarity transformations of a configuration matrix $X$ are the set of translated, rotated and isotropically rescaled $X$, i.e.

$$\{ \beta X \Gamma + 1_k \gamma^T : \beta \in \mathbb{R}^+, \Gamma \in SO(m), \gamma \in \mathbb{R}^m \}, \quad (3.5)$$

where $\beta \in \mathbb{R}^+$ is the scale, $\Gamma$ is a rotation matrix and $\gamma$ is a translation $m$-vector.

The rigid-body transformations of a configuration matrix $X$ are the set of translated and rotated $X$, i.e.

$$\{ X \Gamma + 1_k \gamma^T : \Gamma \in SO(m), \gamma \in \mathbb{R}^m \}, \quad (3.6)$$

where $\Gamma$ is a rotation matrix and $\gamma$ is a translation $m$-vector.

We could consider the shape of $X$ as the equivalence class of the full set of similarity transformations of a configuration. Alternatively we could filter out the similarity transformations from a configuration in a systematic manner. We shall adopt the latter approach.

We are considering non complete coincident landmarks.

3.3.2 Filtering translation

We can remove the set of similarity transformations one at a time in order to obtain a shape representation. Translation is the easiest to remove as it can be achieved by pre-multiplying $X$ by the centring matrix of Equation 3.3.

We write

$$X_C = CX \quad (3.7)$$
and we refer to $X_C$ as the **Centred landmarks**.

### 3.3.3 Pre-shape

In order to eliminate the size information it is necessary to standardize for size. This is achieved by dividing through our notion of size. We choose the centroid size (see Equation 3.2) which is also given by

$$\|CX\| = \sqrt{\text{trace} \left( X^T C X \right)} = S(x),$$  \hspace{1cm} (3.8)

Note that $S(x) > 0$ because we do not allow complete coincidence of landmarks. The pre-shape of a configuration matrix $X$ has all information about location and scale removed.

The **centered pre-shape** of a configuration matrix $X$ is given by

$$Z_C = \frac{CX}{\|CX\|}$$

(3.9)

which is invariant under the translation and scaling of the original configuration.

Note that although $Z_C$ is a $k \times m$ matrix, the pre-shape space has real dimension $(k - 1)m - 1$ since $m$ dimensions have been removed when filtering out translation and 1 when standardizing for size (see Dryden and Mardia [16]).

The **pre-shape space** is the space of all possible pre-shapes. Formally, the pre-shape space $S^k_m$ is the orbit space of the non-coincident $k$ point set configurations in $\mathbb{R}^m$ under the action of translation and isotropic scaling.

The pre-shape space $S^k_m \equiv S^{(k-1)m-1}$ is a hypersphere of unit radius in $(k - 1)m - 1$ real dimensions, since $\|Z_C\| = 1$. The term ‘pre-shape’ signifies that we are one step away from shape - rotation still has to be removed.

### 3.3.4 Shape

In order to also remove rotation information from the configuration we identify all rotated versions of the pre-shape with each other, and this set or equivalence class is the shape of $X$. An alternative definition of the shape of $X$ is

The **shape** of a configuration matrix $X$ is all the geometrical information about $X$ that is invariant under location, rotation and isotropic scaling (Euclidean similarity transformations). The shape can be represented by the set $[X]$ given by

$$[X] = \{Z\Gamma : \Gamma \in SO(m)\},$$  \hspace{1cm} (3.10)
where \( SO(m) \) is the special orthogonal group of rotations and \( Z \) is the pre-shape of \( X \).

The **shape space** is the set of all possible shapes. Formally, the shape space \( \Sigma_m^k \) is the orbit space of the non-coincident \( k \) point set configurations in \( \mathbb{R}^m \) under the action of the Euclidean similarity transformations.

The dimension of the shape space is

\[
M = km - m - 1 - \frac{m(m-1)}{2},
\]

and this can be simply seen as we initially have \( km \) coordinates and then must lose \( m \) dimensions for location, one dimension for uniform scale and \( \frac{1}{2}m(m-1) \) for rotation.

In the language of geometry, the rotated \( Z \) on the pre-shape sphere is called a fibre of the pre-shape space \( S_m^k \). Fibres on the pre-shape sphere correspond one to one with shapes in the shape space, and so we can think of a fibre as representing the shape of a configuration. The pre-shape sphere is partitioned into fibres by the rotation group \( SO(m) \) and the fibre is the orbit of \( Z \) under the action of \( SO(m) \). The fibres do not overlap. We write \( \Sigma_m^k = S_m^k / SO(m) \) and say that \( \Sigma_m^k \) is the quotient space of \( S_m^k \) under the action of \( SO(m) \). Later in Figure 3.2 we see a diagrammatic view of two fibres on the pre-shape sphere.

### 3.3.5 Size-and-shape: Removing location and rotation

In some situations we may need to keep the size information. Then we should remove location and rotation but not scale. Then we have the size-and-shape of \( X \).

**The size-and-shape of a configuration matrix** \( X \) **is all the geometrical information about** \( X \) **that is invariant under location and rotation (rigid-body transformations), and this can be represented by the set** \( [X]_S \) **given by**

\[
[X]_S = \{X_C \Gamma : \Gamma \in SO(m)\},
\]

where \( X_C \) are the centred coordinates of Equation 3.7. The space of all size-and-shapes is called the **size-and-shape space** and is denoted by \( S\Sigma_m^k \), for \( k \) points in \( m \) dimensions. The size-and-shape space is the orbit space of the configuration space under the action of translation and rotation.

If size is removed from the size-and-shape (e.g. by rescaling to unit centroid size), then we obtain the shape of \( X \),

\[
[X] = [X]_S / S(X) = \{Z \Gamma : \Gamma \in SO(m)\},
\]

as in Equation 3.10.
3.4 Distances

In this Section we discuss various choices of distance in the shape space. All of them are based on Procrustes distances. We also introduce an alternative shape coordinate system, the tangent space that is useful for analysing the main modes of shape variation.

3.4.1 Procrustes distances

A concept of distance between two shapes is required to fully define the non-Euclidean shape metric space. We shall primarily concentrate on the full Procrustes distance.

Consider two configuration matrices from $k$ points in $m$ dimensions $X_1$ and $X_2$ with pre-shapes $Z_1$ and $Z_2$. We minimize over rotations and scale to find the closest Euclidean distance between $Z_1$ and $Z_2$.

The full Procrustes distance between $X_1$ and $X_2$ is

$$d_F(X_1, X_2) = \inf_{\Gamma \in SO(m), \beta \in \mathbb{R}} \| Z_2 - \beta Z_1 \Gamma \|, \quad (3.14)$$

where $Z_r = \frac{CX_r}{\| CX_r \|}$, $r = 1, 2$.

The full Procrustes distance is

$$d_F(X_1, X_2) = \left( 1 - \left( \frac{m}{\sum_{i=1}^{m} \lambda_i} \right)^2 \right)^{\frac{1}{2}}, \quad (3.15)$$
where \( \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_{m-1} \geq |\lambda_m| \) are the square roots of the eigenvalues of \( Z_1^T Z_2 Z_1^T Z_1 \). The minimizing rotation is given by

\[
\hat{\Gamma} = U V^T, \tag{3.16}
\]

where \( U, V \in SO(m) \) and \( Z_2^T Z_1 = V \Lambda U^T \) with \( \Lambda = \text{diag} (\lambda_1, \lambda_2, \ldots, \lambda_m) \).

The minimizing scale is

\[
\hat{\beta} = \sum_{i=1}^{m} \lambda_i. \tag{3.17}
\]

**Proof:** To obtain the minimizing rotation we follow Kendall [32]. Note that

\[
d_2^p (X_1, X_2) = \inf_{\Gamma \in SO(m), \beta} \text{trace} \left\{ (Z_2 - \beta Z_1 \Gamma)^T (Z_2 - \beta Z_1 \Gamma) \right\}
\]

\[
= \inf_{\beta} \left( \| Z_2 \|^2 + \beta^2 \| Z_1 \|^2 - 2\beta \sup_{\Gamma \in SO(m)} \text{trace} (Z_2^T Z_1 \Gamma) \right), \tag{3.18}
\]

and \( \| Z_1 \| = 1 = \| Z_1 \| \). We wish to find the supremum of \( \text{trace} (Z_2^T Z_1 \Gamma) \) over \( \Gamma \in SO(m) \). First, consider a singular value decomposition of \( Z_1^T Z_2 \) given by

\[
Z_1^T Z_2 = V \Lambda U^T, \tag{3.19}
\]

where \( U, V \in SO(m), \Lambda = \text{diag} (\lambda_1, \ldots, \lambda_m) \) with \( \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_{m-1} \geq |\lambda_m| \). Note that \( \lambda_m < 0 \) if and only if \( \det (Z_2^T Z_1) < 0 \). Hence, the last term in the right-hand side of Equation 3.18 is equivalent to

\[
-2\beta \sup_{R \in SO(m)} \text{trace} (R \Lambda) = -2\beta \sup_{R \in SO(m)} \sum_{i=1}^{m} r_{ii} \lambda_i, \tag{3.20}
\]

where \( (r_{11}, \ldots, r_{mm}) \) are the diagonals of \( R \in SO(m) \). Now the set of diagonals of \( R \in SO(m) \) is a compact convex set with extreme points

\[
\{ (\pm 1, \pm 1, \ldots, \pm 1) \}
\]

with an even number of minus signs. Hence, it is clear in our case that the supremum is achieved when \( r_{ii} = 1, i = 1, \ldots, m \). Hence

\[
d_2^p (X_1, X_2) = \inf_{\beta} \left( 1 + \beta^2 - 2\beta \sum_{i=1}^{m} \lambda_i \right). \tag{3.21}
\]

By differentiation we have

\[
\hat{\beta} \sum_{i=1}^{m} \lambda_i,
\]

as required. Substituting \( \hat{\beta} \) into Equation 3.21 leads to the correct expression for the full Procrustes distance, as required. It is also clear that the minimizing rotation is found at \( \hat{\Gamma} = U V^T \) since

\[
\text{trace} (Z_2^T Z_1 \Gamma) = \text{trace} (V \Lambda U^T U V^T) = \text{trace} (\Lambda). \quad \square
\]
Figure 3.2: A diagrammatic simplistic view of two fibres $[X_1]$ and $[X_2]$ on the pre-shape sphere, which correspond to the shapes of the original configuration matrices $X_1$ and $X_2$ which have pre-shapes $Z_1$ and $Z_2$. Also displayed are the smallest great circle $\rho$ and chordal distances $d_P$ between the fibres.

### 3.4.2 Alternative distances

Alternative distances in shape space could be suggested. In Figure 3.2 we see a diagrammatic view of the pre-shape sphere. Since the pre-shape sphere is a hypersphere embedded in $\mathbb{R}^{(k-1)m}$ we could consider familiar distances between two points on a sphere, such as the great circle distance or the chordal (Euclidean) distance. Since the shapes of configurations are represented by fibres on the pre-shape sphere, we can define the distance between two shapes as the closest distance between the fibres on the pre-shape sphere. In figure 3.2 two minimum distances have been drawn between the fibres (shapes), $\rho$ is the closest great circle distance and $d_P$ the closest chordal distance.

The **partial Procrustes distance** $d_P$ is obtained by matching the pre-shapes $Z_1$ and $Z_2$ of $X_1$ and $X_2$ as closely as possible over rotations, but not scale. So,

$$d_P(X_1, X_2) = \inf_{\Gamma \in SO(m)} \| Z_2 - Z_1 \Gamma \|,$$

where the $Z_j$ are the pre-shapes.

The partial Procrustes distance is given by

$$d_P(X_1, X_2) = \sqrt{2} \left( 1 - \sum_{i=1}^{m} \lambda_i \right)^{\frac{1}{2}}. \quad (3.22)$$
Figure 3.3: Section of the pre-shape sphere, illustrating the relationship between the Procrustes distances $d_F$, $d_P$ and $\rho$.

**Proof:** By keeping $\beta = 1$ fixed throughout the proof of Equation 3.15, and just minimizing over $\Gamma$.

Note the optimal rotation is the same whether or not scaling is in the minimization.

The **Procrustes distance** $\rho(X_1, X_2)$ is the closest great circle distance between $Z_1$ and $Z_2$ on the pre-shape sphere, where $Z_j$ are the pre-shapes. The minimization is carried out over rotations.

From trigonometry one can see that the Procrustes distance is

$$\rho(X_1, X_2) = \arcsin(d_P(X_1, X_2)/2) = \arccos \left( \frac{\Sigma_{i=1}^m \lambda_i}{2} \right). \quad (3.23)$$

In figure 3.3 we see a cross-section of the pre-shape sphere illustrating the relationships between $d_F$, $d_P$ and $\rho$. Indeed

$$d_F(X_1, X_2) = \sin \rho,$$

$$d_P(X_1, X_2) = 2 \sin (\rho/2).$$
Note that $\rho$ can be considered as the smallest angle between the complex vectors $Z_1$ and $Z_2$ over rotations of $Z_1$ and $Z_2$.

The minimization over rotations in the Procrustes distance calculations could also be obtained using calculus (see Dryden and Mardia [16] section 5.2.1). In particular, $\rho(X_1, X_2)$ is the geometrically natural distance inherited from the projection of the fibres on the pre-shape sphere to points in the shape space. This projection is termed a Riemannian submersion in geometry.

For shapes which are close together there is very little difference between the shape distances, since

$$d_P = d_F + O\left( d_F^3 \right), \quad \rho = d_F + O\left( d_F^3 \right).$$

Consequently for many practical datasets with small variability there is very little difference in the analyses when using different Procrustes distances. However, the distinction between the distances is worth making and the terminology is summarized in Table 3.1.

<table>
<thead>
<tr>
<th>Distance</th>
<th>Notation</th>
<th>Formula</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full Procrustes distance</td>
<td>$d_F$</td>
<td>$\left{ 1 - \left( \sum_{i=1}^{m} \lambda_i \right)^2 \right}^{1/2}$</td>
<td>$0 \leq d_F \leq 1$</td>
</tr>
<tr>
<td>Partial Procrustes distance</td>
<td>$d_P$</td>
<td>$\sqrt{2(1 - \sum_{i=1}^{m} \lambda_i)^{1/2}}$</td>
<td>$0 \leq d_P \leq \sqrt{2}$</td>
</tr>
<tr>
<td>Procrustes distance</td>
<td>$\rho$</td>
<td>$\arccos\left( \sum_{i=1}^{m} \lambda_i \right)$</td>
<td>$0 \leq \rho \leq \pi/2$</td>
</tr>
</tbody>
</table>

Table 3.1: Procrustes distances in the shape space

We illustrate the geometrical steps involved in obtaining the full and partial Procrustes distance in figure 3.4. The first row shows the original figure $X_1$ (left), the centered $X_1$ (middle) and the rescaled $X_1$, which we call $Z_1$ (right). The second row shows the original figure $X_2$ (left), the centered $X_2$ (middle) and the rescaled $X_1$, which we call $Z_1$ (right). In the third row (left) $Z_2$ is rotated to $Z_1$ to minimize the sum of squared distances between pairs of landmarks - the partial Procrustes distance is the Euclidean distance between these fitted configurations. In the third row (middle) $Z_2$ is rotated and rescaled to $Z_1$ to minimize the sum of squared distances between pairs of landmarks - the full Procrustes distance is the Euclidean distance between these fitted configurations.

### 3.4.3 Tangent space coordinates

The tangent space is the linearized version of the shape space in the vicinity of a particular point of shape space (the pole of the tangent projection). The pole is usually chosen to be an average shape obtained from the dataset of interest, and hence this choice of coordinates depends on the dataset under study.

The Euclidean distance in the tangent space to the shape space is a good approximation to the Procrustes distances $d_F$, $d_P$ and $\rho$ in shape space in the vicinity of the pole. In fact can be proven that the Euclidean distance in the
Figure 3.4: The geometry of Procrustes fits in calculating Procrustes distances. The first two rows show each configuration (left), centered (middle) and scaled (right). The last row shows the partial Procrustes fit (left) and full Procrustes fit (middle) of the centered pre-shape from configuration 1 (dashed line) onto the centered pre-shape of configuration 2 (solid line)
tangent space to the pole of the projection is exactly the same as the full Pro-
crutes distance to the pole. So if most of the objects in a dataset are quite close
in shape, then using the Euclidean distance in the tangent space will be a good
approximation to the shape distances in the shape space. Hence, in practical
shape analysis we shall see that the tangent space is extremely important and
useful.

We now give an approximation to the full Procrustes tangent space (Goodall
[22]; Cootes [10]), which can be formulated when the pole is the Procrustes
mean. After observations \( w_1, \ldots, w_n \) have been superimposed by generalized
Procrustes analysis to \( w_1^P, \ldots, w_n^P \) consider the Procrustes residuals
\[
r_i = w_i^P - \left( \frac{1}{n} \sum_{i=1}^{n} w_i^P \right), \quad i = 1, \ldots, n.
\]

The Procrustes residuals are approximate tangent coordinates.

3.5 General Procrustes Methods

This Section outlines the methods based on Procrustes superimposition, which
are very useful tools for analysing landmark data. Procrustes methods were
seen to be useful for assessing distances between shapes in Section 3.4. In this
Section we provide a more complete treatment of Procrustes methods suitable
for two and higher dimensional shape analysis.

Procrustes methods are useful for estimating an average shape and for ex-
ploring the structure of shape variability in a dataset.

3.5.1 Introduction

Procrustes analysis involves matching configurations with similarity transfor-
mations to be as close as possible in Euclidean distance.

When we have two configurations we use ordinary Procrustes analysis. Oth-
wise, when we have more than two configurations we use the technique of
generalized Procrustes analysis (GPA). Both techniques allow for matching
the configurations as close as possible using similarity transformations but also for
obtaining an average shape.

After an estimate of mean shape has been obtained we wish to analyze
the shape variability using Principal Components Analysis (PCA) of the shape
differences around the mean.

3.5.2 Full ordinary Procrustes analysis

Consider that we have two configurations \( X_1 \) and \( X_2 \), both \( k \times m \) matrices of
\( k \) points in \( m \) dimensions, and we want to match them as close as possible, up
to similarity transformations. We assume in this section, and without loss of
generality, that the configuration matrices have been centered using equation
The method of full ordinary Procrustes analysis (full OPA) involves least squares matching of two configurations using the full set of similarity transformations. Estimation of the parameters $\gamma$, $\Gamma$ and $\beta$ is carried out by minimizing the squared Euclidean distance

$$D_{\text{OPA}}^2(X_1, X_2) = \| X_2 - \beta X_1 \Gamma - 1_k \gamma^T \|^2,$$  
(3.25)

where $\| X \| = \{\text{trace} (X^T X)\}^{1/2}$ is the Euclidean norm, $\Gamma$ is an $(m \times m)$ rotation matrix ($\Gamma \in \text{SO}(m)$), $\beta > 0$ is a scale parameter and $\gamma$ is an $(m \times 1)$ location vector. The minimum of Equation 3.25 is written as $\text{OSS}(X_1, X_2)$, which stands for Ordinary (Procrustes) Sum of Squares.

In section 3.4.1 we were interested in the minimum value of an expression similar to Equation 3.25 except that $X_1$ and $X_2$ were of unit size.

The full ordinary Procrustes solution to the minimization of Equation 3.25 is given by $(\hat{\gamma}, \hat{\beta}, \hat{\Gamma})$ where

$$\hat{\gamma} = 0$$
(3.26)

$$\hat{\Gamma} = UV^T$$
(3.27)

where

$$X_2^T X_1 = \| X_1 \| \| X_2 \| V \Lambda U^T, U, V \in \text{SO}(m)$$
(3.28)

with $\Lambda$ a diagonal $m \times m$ matrix of positive elements except possibly the last element defined in section 3.4.1. Furthermore,

$$\hat{\beta} = \frac{\text{trace} (X_2^T X_1 \hat{\Gamma})}{\text{trace} (X_1^T X_1)}$$
(3.29)

and

$$\text{OSS}(X_1, X_2) = \| X_2 \| \| \frac{\| X_2 \|^2}{\sin^2 \rho(X_1, X_2)} \| \rho(X_1, X_2),$$
(3.30)

where $\rho(X_1, X_2)$ is the Procrustes distance of equation 3.23.

**Proof:** We wish to minimize

$$D_{\text{OPA}}^2(X_1, X_2) = \| X_2 - \beta X_1 \Gamma - 1_k \gamma^T \|^2 = \text{trace} (\| X_2 \|^2 + \beta^2 \| X_1 \|^2 - 2\beta X_2^T X_1 \Gamma) + k\gamma^T \gamma,$$

where $X_1$ and $X_2$ are centered. It is simple to see that we must take $\hat{\gamma} = 0$. If
\[ Z_i = CX_i / \| CX_i \|, \ i = 1, 2, \]

are the pre-shapes of \( X_i \), then we need to minimize

\[ \text{trace} \left( \| X_2 \|^2 + \beta^2 \| X_1 \|^2 - 2\beta \| X_1 \| \| X_2 \| Z_2^T Z_1 \Gamma \right) \]

and so from section 3.4.1 we find the minimizing \( \Gamma \) from equation 3.16. Differentiating with respect to \( \beta \) we obtain:

\[ \frac{\delta D^2_{OPA}}{\delta \beta} = 2\beta \text{trace} \left( X_1^T X_1 \right) - 2\text{trace} \left( \| X_1 \| \| X_2 \| Z_2^T Z_1 \hat{\Gamma} \right). \]

Hence,

\[ \hat{\beta} = \left( \frac{\| X_2 \| \text{trace} \left( Z_2^T Z_1 \hat{\Gamma} \right)}{\| X_1 \| \text{trace} (\Lambda)} \right) = \frac{\| X_2 \|}{\| X_1 \|} \text{cos } \rho (X_1, X_2). \] (3.31)

Substituting \( \hat{\gamma} \), \( \hat{\Gamma} \) and \( \hat{\beta} \) into Equation 3.25 leads to

\[ \text{OSS} (X_1, X_2) = \| X_2 \|^2 + \hat{\beta}^2 \| X_1 \|^2 - 2\hat{\beta} \| X_1 \| \| X_2 \| \text{cos } \rho \]

and so the result of Equation 3.30 follows. \( \square \)

### 3.5.3 Generalized Procrustes Analysis

Consider the case that we have \( n \geq 2 \) configuration matrices \( X_1, \ldots, X_n \). These configurations could be a random sample from a population with mean shape \( \mu \) which we wish to estimate from the sample with an average shape \( [\mu] \).

Let us consider three perturbation models for the \( k \times m \) configuration matrices \( X_i \),

\[ X_i = \mu + E_i \] (3.32)

\[ X_i = (\mu + E_i) \Gamma_i + 1_k \gamma_i^T \] (3.33)

\[ X_i = \beta_i (\mu + E_i) \Gamma_i + 1_k \gamma_i^T, \] (3.34)

where \( E_i \) are zero mean \( k \times m \) independent random error matrices, \( \mu \) is the \( k \times m \) matrix of the mean configuration and \( \beta_i, \Gamma_i \) and \( \gamma_i \) are nuisance parameters for scale, rotation and translation.

We can estimate \( \mu \) directly under the error model 3.32, although this model is rarely applicable in practice.

We cannot estimate directly \( \mu \) under models 3.33 and 3.34 because of what we have called nuisance parameters which correspond to rotations and translations in 3.33 and scalings, rotations and translations in 3.34. The parameters
(Γ_i, β_i, γ_i) have been termed ‘nuisance parameters’ by Goodall ([10]) because they are not parameters of primary interest in shape analysis.

Under models 3.33 and 3.34 we must first filter out these nuisance parameters in order to get a mean estimator [μ].

We can estimate the shape [μ] or size-and-shape [μ]s under model 3.33 and we can estimate shape [μ] under model 3.34.

For size-and-shape analysis our objects need to be commensurate in scale, which means that there must be no nuisance on the scale information thus, model 3.33 is appropriate. For shape analysis our objects need not be commensurate in scale, and so model 3.33 or 3.34 can be used.

The method of full generalized Procrustes analysis (full GPA) involves translating, rescaling and rotating the configurations relative to each other so as to minimize a total sum of squares, and the procedure is appropriate under model 3.33 or 3.34. We minimize a quantity proportional to the sum of squared norms of pairwise differences,

\[ G(X_1, \ldots, X_n) = \frac{1}{n} \sum_{i=1}^{n} \sum_{j=i+1}^{n} \| (\beta_i X_i \Gamma_i + 1 k \gamma_i^T) - (\beta_j X_j \Gamma_j + 1 k \gamma_j^T) \|^2 \]  

subject to a constraint on the size of the average,

\[ S(\bar{X}) = 1, \]  

where \( \Gamma_i \in SO(m), \beta_i > 0, \| X \| = \sqrt{\text{trace}(X^TX)} \) and \( S(X) \) is the centroid size and the average configuration is

\[ \bar{X} = \frac{1}{n} \sum_{i=1}^{n} (\beta_i X_i \Gamma_i + 1 k \gamma_i^T). \]  

We write \( G(X_1, \ldots, X_n) \) for the minimum of Equation 3.35, subject to the constraint of Equation 3.36, and \( G(X_1, \ldots, X_n) \) is called the generalized (Procrustes) sum of squares.

Full generalized Procrustes matching consists on the superimposition of all configurations placed ‘on top of each other’ in optimal positions by translating, rotating and rescaling each figure so as to minimize the sum of squared Euclidean distances. The constraint of Equation 3.36 prevents the \( \beta_i \) from all being close to 0. Note that

\[ G(X_1, \ldots, X_n) = \inf_{\beta_i, \Gamma_i, \gamma_i} \frac{1}{n} \sum_{i=1}^{n} \sum_{j=i+1}^{n} \| (\beta_i X_i \Gamma_i + 1 k \gamma_i^T) - (\beta_j X_j \Gamma_j + 1 k \gamma_j^T) \|^2 \]

\[ = \inf_{\beta_i, \Gamma_i, \gamma_i} \sum_{i=1}^{n} \| (\beta_i X_i \Gamma_i + 1 k \gamma_i^T) - \frac{1}{n} \sum_{j=1}^{n} (\beta_j X_j \Gamma_j + 1 k \gamma_j^T) \|^2. \]

\[ = \inf_{\beta_i, \Gamma_i, \gamma_i} \sum_{i=1}^{n} \| (\beta_i X_i \Gamma_i + 1 k \gamma_i^T) - \bar{X} \| \]
It follows that the minimization can be viewed as another constrained estimation problem, where the mean shape \( \mu \) is to be estimated, i.e.

\[
G(X_1, \ldots, X_n) = \inf_{\mu: S(\mu) = 1} \sum_{i=1}^{n} \text{OSS}(X_i, \mu) = \inf_{\mu: S(\mu) = 1} \sum_{i=1}^{n} \sin^2 \rho(X_i, \mu). \tag{3.38}
\]

The full Procrustes fit (or full Procrustes coordinates of each of the \( X_i \)) is given by

\[
X_i^P = \hat{\beta}_i X_i \hat{\Gamma}_i + 1_k \hat{\gamma}_iT, \quad i = 1, \ldots, n, \tag{3.39}
\]

where \( \hat{\Gamma}_i \in SO(m) \) (rotation matrix), \( \hat{\beta}_i > 0 \) (scale parameter), \( \hat{\gamma}_iT \) (location parameters), \( i = 1, \ldots, n \) are the minimizing parameters.

An algorithm to estimate the transformation parameters \( (\gamma_i, \beta_i, \Gamma_i) \) is described below.

The full Procrustes estimate of mean shape (full Procrustes mean) is given by \([\hat{\mu}]\), where

\[
\hat{\mu} = \arg \inf_{\mu: S(\mu) = 1} \sum_{i=1}^{n} \sin^2 \rho(X_i, \mu) = \arg \inf_{\mu: S(\mu) = 1} \sum_{i=1}^{n} d^2_F(X_i, \mu). \tag{3.40}
\]

Note that

\[
G(X_1, \ldots, X_n) = \inf_{\mu: S(\mu) = 1} \sum_{i=1}^{n} \sin^2 \rho(X_i, \mu).
\]

The point in shape space corresponding to the arithmetic mean of the Procrustes fits,

\[
\bar{X} = \frac{1}{n} \sum_{i=1}^{n} X_i^P,
\]

has the same shape as the full Procrustes mean.

**Proof:** The result follows because we are minimizing sums of Euclidean square distances in generalized Procrustes analysis. The minimum of

\[
\sum \| X_i^P - \mu \|^2
\]

over \( \mu \) is given by \( \hat{\mu} = n^{-1} \sum_i X_i^P \).

Hence, once a collection of objects has been matched into optimal full Procrustes position with respect to each other, calculation of the full Procrustes mean shape consists in taking arithmetic means of each coordinate. Full generalized Procrustes analysis is analogous to minimizing sums of squared distances in the shape space \( d^2_F \) defined in equation 3.15.
Although there exists an explicit eigenvector solution to the case of \( m = 2 \) dimensions, we are using the general iterative approach applicable to any number of dimensions. This algorithm, developed by Gower [23] and modified by Ten Berge [2] is as follows:

**Algorithm: GPA**

1. **Translations.** Centre the configurations to remove location. Initially let
   \[
   X^P_i = X_i, \ i = 1, \ldots, n.
   \]

2. **Rotations.** For the \( i \)th configuration let
   \[
   \bar{X}_{(i)} = \frac{1}{n-1} \sum_{j \neq i} X^P_j,
   \]
   then the new \( X^P_i \) is taken to be the ordinary Procrustes superimposition, involving only rotation, of the old \( X^P_i \) on \( \bar{X}_{(i)} \). The \( n \) figures are rotated in turn. This process is repeated until the Procrustes sum of squares of Equation 3.35 cannot be reduced further.

3. **Scaling.** Let \( \Sigma \) be the \( n \times n \) correlation matrix of the \( \text{vec} \left( X^P_i \right) \) (with the usual roles of variable and observation labels reversed) with eigenvector \( \phi_k = (\phi_{k1}, \ldots, \phi_{kn})^T \) corresponding to the largest eigenvalue. Then from Ten Berge [2] take
   \[
   \hat{\beta}_i = \left( \frac{\sum_{k=1}^n \| X^P_k \|^2}{\| X^P_i \|^2} \right)^{1/2} \phi_{ki},
   \]
   which is repeated for all \( i \).

4. Repeat steps 2 and 3 until the Procrustes sum of squares of Equation 3.35 cannot be reduced further.

The algorithm usually converges quickly.

When \( n = 2 \) objects are available we can consider OPA or GPA to match them. The advantage of using GPA is that the matching procedure is symmetrical in the ordering of the objects, i.e. GPA of \( X_1 \) and \( X_2 \) is the same as GPA of \( X_2 \) and \( X_1 \). On the other hand, OPA is not symmetrical in general as seen in section 3.5.2, unless the objects have the same size.

In Figure 3.5(a) we see a set of \( n = 5 \) synthetic configurations of \( k = 6 \) landmarks in \( m = 2 \) dimensions in their original dispositions. The objects
represent a square with some kind of tail at one of the corners. Both the aspect of the square and the length of the tail are variable among the examples. Figure 3.5(b) shows the results after full GPA superimposition. Figure 3.5(c) shows the full Procrustes estimate of mean shape, which corresponds to the arithmetic mean of each coordinate of the configurations after full GPA superimposition (as seen in the Proof of Equation 3.40).

### 3.5.4 Partial Procrustes Analysis

Partial Procrustes involves matching configurations by translating and rotating them as opposed to full Procrustes which involves matching them by the full set of similarity transformations. Partial Procrustes is suitable when estimating size-and-shape.

#### Ordinary partial Procrustes

One may be interested in estimating size-and-shape, in which case it is not of interest to initially scale to unit size. The shapes must be commensurate in scale.

**Ordinary partial Procrustes analysis** involves superimposition of two configurations by only translation and rotation. This does not necessarily require the figures to be of unit size. Minimization of the following expression is required.

\[
\| X_2 - X_1 \Gamma - 1_k \gamma^T \|^2.
\]

The minimizing translation and rotation are used as in the full Procrustes case (Equations 3.26 and 3.27). The minimum of the expression is

\[
OSS_p(X_1, X_2) = \text{trace} \left( X_1^T X_1 \right) + \text{trace} \left( X_2^T X_2 \right) - 2 \| X_1 \| \| X_2 \| \cos \rho(X_1, X_2).
\]

If the two configurations are of unit size, then equation 3.41 is equal to \(d_P^2(X_1, X_2)\), the square of the partial Procrustes distance.

#### Generalized partial Procrustes

When we have more than two objects we can match them together using translation and rotation alone. It is not necessary to pre-scale to unit size. In that case we are estimating the size-and-shape. The objects must be commensurate in scale and so model 3.33 is appropriate. We minimize

\[
GP = \inf_{\Gamma, \gamma, \mu} \sum_{i=1}^n \| X_i \Gamma_i + 1_k \gamma_i^T - \mu \|^2
\]

\[
= \inf_{\mu} \sum_{i=1}^n \left\{ \| \mu \|^2 + \| X_i \|^2 - 2 \| X_i \| \| \mu \| \cos \rho(X_i, \mu) \right\},
\]

(3.42)
Figure 3.5: Results of full GPA superimposition on a synthetic set of example configurations representing a square with a tail in one of the corners. Full Procrustes estimate of mean shape is also shown.
which follows since

$$\text{trace} \left( \mu^T X_i \Gamma_i \right) = \| X_i \| \| \mu \| \cos \rho.$$ 

If all the configurations have unit size, then it can be seen that the estimator of $[\mu]$ is obtained by minimizing sums of squared chordal distances on the pre-shape sphere, and so

$$G_p = \inf_{\mu} \sum_{i=1}^{n} d^2_P (X_i, \mu)$$

$$= \inf_{\mu} \sum_{i=1}^{n} 2 \{ 1 - \cos \rho (X_i, \mu) \}.$$  (3.43)

Full Generalized Procrustes Analysis (full GPA) algorithm was designed to do the superimposition of a set of centered configurations through rotations and scaling, thus obtaining an estimate of mean shape (full Procrustes estimate of mean shape). Consider the case that we are interested in retaining the size information. Then, it is logical that such superimposition must be done through rotations alone. Is for that reason that we propose a variant on the full GPA algorithm of Section 3.5.3 useful for obtaining the size-and-shape representation of a set of configurations. Note that they need to be commensurate in scale, therefore is useful for perturbation models as in Equation 3.33. It is based on the full GPA algorithm but without the scaling step. Hence, the procedure for this new algorithm consists on the steps 1 and 2 of the full GPA algorithm. We call it partial GPA since it does retain the size information and it can be applied to any number of configurations. This new algorithm is supposed to converge more quickly that the full GPA since it consists on a version with less restrictions, i.e. it does not need to iterate through rotations and scaling until convergence.

Figure 3.5.4 shows an example of superimposition using the partial GPA algorithm described above. We use the set of $n = 5$ synthetically generated configurations of $k = 6$ landmarks shown in Figure 3.6(a). Figure 3.6(b) shows the results after partial GPA superimposition. Figure 3.6(c) shows the partial Procrustes estimate of mean shape.
Figure 3.6: Results of partial GPA superimposition on a synthetic set of example configurations. Partial Procrustes estimate of mean shape is also shown.
Chapter 4

Modelling Shape Variability

When one has moved all the shapes into a common reference frame with all the
Euclidean similarity transformations filtered-out and has obtained an average
shape, is of great interest to analyze the main modes of variation around this
average shape. Principal Components Analysis on the tangent space coordinates
provides very effective means for extracting such modes.

The aim of this Chapter is to describe the process of building a model of
shape variation, that is to extract the main modes of variation of a class of
shapes. We do it for both linear and non-linear case.

In Section 4.1 we give an overview of Principal Components Analysis, the
main technique involved in extracting the structure of shape variability. In
Section 4.2 we describe methods for extracting linear modes of shape variability
while in Section 4.3 we do it for non-linear ones.

4.1 Principal Components Analysis

Consider $X$ a random variable with covariance matrix $\Sigma_X$,

$$\Sigma_X = (X - \mu_X)(X - \mu_X)^T,$$

where $\mu_X = E\{X\}$ is the mean.

The objective is to find a transformation matrix $P$ such that eliminates the
correlation. That is

$$Y = P^T X,$$

where $\Sigma_Y$, the covariance matrix of $Y$ is diagonal. Or
\[ \Sigma_Y = P^T \Sigma_X P \quad (4.1) \]

where \( \Sigma_Y \) is a diagonal matrix.

Let \( \Sigma_Y = D \) be an \((n \times n)\) diagonal matrix; and let \( \Sigma_X = A \) the \((n \times n)\) matrix to be diagonalized.

From Equation 4.1 we have

\[ D = P^T AP \Rightarrow AP = PD \quad (4.2) \]

where

\[
D = \begin{bmatrix}
\lambda_1 & 0 & \cdots & 0 \\
0 & \lambda_2 & \cdots & 0 \\
\vdots & \ddots & \ddots & \vdots \\
0 & 0 & \cdots & \lambda_n
\end{bmatrix}
\]

that is the same than finding a set of vectors \( P \) such that when we apply the transformation \( A \), they do not change in direction but only in scale:

\[
[A_1|A_2|\ldots|A_n][P_1|P_2|\ldots|P_n] = [P_1|P_2|\ldots|P_n]
\]

\[
= [\lambda_1 P_1|\lambda_2 P_2|\ldots|\lambda_n P_n]
\]

where \( A = [A_1|A_2|\ldots|A_n] \) and \( P = [P_1|P_2|\ldots|P_n] \) \((A_i, P_i \text{ are } (n \times 1) \text{ vectors})\).

We must find a set of vectors \( v \) such that:

\[ Av = \lambda v \Rightarrow (A - \lambda I) v = 0 \quad (4.3) \]

In order to find any \( v \neq 0 \) that satisfy Equation 4.3, then

\[ det (A - \lambda I) \neq 0 \]

since, at least one vector from \((A - \lambda I)\) must be linearly dependent.

We call

\[ det (A - \lambda I), \quad (4.4) \]

the characteristic equation of \( A \); and
\((A - \lambda I)\), \hspace{1cm} (4.5)

the characteristic matrix of \(A\).

We find the \(\lambda_i\) that satisfy Equation 4.4, and next we get the vectors \(v_i\) associated to each \(\lambda_i\). These \(v_i \equiv P_i\) are the eigenvectors of \(A\), and \(P = [P_1|P_2|\ldots|P_n]\) is the eigenvectors matrix.

**Properties of \(P\)**

1. \(P\) is composed by linearly independent vectors. Hence \(P\) form a basis for \(\mathbb{R}^n\)
2. An \((n \times n)\) matrix \(A\) has \(n\) eigenvectors. Then the eigenvectors of a \((n \times n)\) matrix form a basis for \(\mathbb{R}^n\).
3. \(P\) is an invertible matrix, since \(P^{-1} = P^T\)

Let \(\Phi = P\) be the eigenvectors matrix of \(\Sigma X\). Let \(\Sigma X\) be the covariance matrix of \(X\).

We use the transformation

\[ Y = \Phi^T X \]

then from Equation 4.2

\[ \Sigma_Y = \Phi^T \Sigma_X \Phi = \Phi^T \Phi \Lambda = \Lambda \]

\[ \Rightarrow \Sigma_Y = \Lambda \]

**Conclusions**

1. Let \(y_i = \phi_i^T X\) \((i = 1, \ldots, n)\)
   As \(\phi_i^T X = \| \phi_i \| \| X \| \cos \theta = \| X \| \cos \theta\) where \(\theta\) is the angle between \(X\) and \(\phi_i\),
   then \(y_i\) is the value of the projection of \(X\) on \(\phi_i\)

   \(Y\) represents \(X\) in a new coordinates system spanned by \(\phi_i\) \((i = 1, \ldots, n)\)

2. \(Y\) is decorrelated since \(\Sigma_Y\) is a diagonal matrix
3. $\Phi$ is an orthogonal transformation since Euclidean Distance is preserved

$$\| Y \|^2 = Y^T Y = X^T \phi \phi^T X = X^T X = \| X \|^2$$

### 4.2 Linear Shape Modelling

The full or partial Procrustes estimate of mean shape provides an effective estimate of mean shape. It is also of great interest to describe variability in shape. Principal Components Analysis (PCA) of the sample covariance matrix in Procrustes tangent space coordinates provides effective means of analysing the main modes of variation in shape. In addition PCA is useful to reduce the dimensionality of a problem as it is in multivariate analysis.

In this Section we give a detailed description of **Point Distribution Models**.

#### 4.2.1 Point Distribution Models

Cootes et al. [10] use PCA to develop *Point Distribution Models* (PDM), which is a PC model for shape and uses Procrustes residuals of Equation 3.24 which are approximately tangent coordinates as seen in Section 3.4.3.

Given $n$ independent configuration matrices $X_1, \ldots, X_n$ the figures are registered to $X_P^1, \ldots, X_P^n$ by any of the GPA procedures described. The corresponding Procrustes mean shape $\hat{\mu}$ is taken to be the estimate of mean shape, which has the same shape as $\bar{X} = \frac{1}{n} \sum_{i=1}^n X_P^i$. The sample covariance matrix is

$$S = \frac{1}{n} \sum_{i=1}^n \text{vec} (X_P^i - \bar{X}) (\text{vec} (X_P^i - \bar{X}))^T$$

and the PCs are the eigenvectors of this matrix, $\phi_j$, $j = 1, \ldots, \min (n - 1, M)$ (M is the dimension of the shape space), with corresponding decreasing eigenvalues $\lambda_j$.

Visualization of the effects of each PC can be carried out through plots of the shapes generated by addition of a displacement vector to the mean shape $\hat{\mu}$.

$$\text{vec} (\bar{X}) + c \lambda_j^{1/2} \phi_j$$

for a range of values of $c$ (e.g. $[-3,3]$). The term $\lambda_j^{1/2}$ corresponds to the standard deviation (s.d.) along $j$th component. Hence, $c$ indicates the number of standard deviations that we displace from the mean.

In datasets where shape variability is small it is often beneficial to magnify the range of $c$ in order to easily visualize the effect of each PC.

By carrying out PCA in the tangent space we are decomposing variability (the total sum of Procrustes distances) into orthogonal components, with each PC successively explaining the highest variability in the data, subject to being orthogonal to the higher PCs.
When strong dependencies between landmarks exist, only a few PCs may be required to explain a high percentage of shape variability. Some PCs may correspond to interpretable aspects of variability (e.g. thickness, bending, shear) although there are often many combined effects in each PC.

Another important feature is that one can get an approximation of a given configuration $X$ by

$$X \approx \text{vec} \left( \bar{X} \right) + \Phi \phi$$  \hspace{1cm} (4.8)

where $\Phi = (\phi_1 | \phi_2 | \ldots | \phi_t)$ contains the $t$ eigenvectors corresponding to the largest eigenvalues, and $b$ is a $t$-dimensional vector generated by

$$b = \Phi^T \text{vec} \left( X - \bar{X} \right),$$  \hspace{1cm} (4.9)

the projection of the configuration $X$ to the eigenspace spanned by the first $t$ eigenvectors (see Fukunaga [21]).

The number of eigenvectors to use $t$ is usually chosen to explain a certain amount of variability (e.g. 98%), so that the residual terms can be considered noise.

For instance figure 4.1(b) show how the distribution of two dimensional vectors represented as points in figure 4.1(a) can be approximated with quite accuracy by the projection values on the first eigenvector, representing the distance from the mean on that direction. This is due to the correlation present between the two dimensions on the original vectors. Thus the two dimensional data is approximated using a model with a single parameter. Similarly shape models controlling many hundred of model points may need only a few parameters to approximate the examples in the original training set.

We applied PDM to the face shapes of Figure 4.2. We used full GPA and partial GPA to obtain the shape and size-and-shape representations respectively. Results of superimposition can be seen in Figures 4.3(a) and 4.3(b). The resulting models are shown in Figure 4.4 for shape, and 4.5 for size-and-shape.

As seen in Figures 4.4(a) and 4.5(a) five eigenvectors are enough to explain the 100% of the variance in both shape and size-and-shape representations. (note that although the shapes are composed by a large amount of landmark points, the training set is composed only by six examples which is not so much).

Figures 4.4(b) and 4.5(b) show how the first mode seems to vary from a surprised expression to a disgusted one. Note how the first mode has captured the relationships between the oberture of the mouth, the form of the face and the position of the eyes and eyebrows. When the mouth is open, the face is stretched and the eyes and eyebrows are more separated and open, giving a surprised expression. Inversely, when the mouth is closed, the face is shrinked and the eyes and eyebrows are more close to each other, thus giving a disgusted expression.
Figure 4.1: Applying PCA on a two dimensional dataset. Any point can be represented by the distance from the mean on the principal axe.

Figure 4.2: Example face shapes
Note also that while in the model built from the shape representation the first mode involves mainly a change in shape, in the model built from the size-and-shape representation this mode encodes also a change in size. This is normal since in the shape representation, the size information has been filtered-out while in the size-and-shape representation it has not.

As seen in Figures 4.4(c) and 4.5(c) the second mode varies from a happy expression to a sad expression. Note that, as in the first mode, this mode has captured again the relationships between the zone of the eyes and the mouth. This correlation brings the eyes to close when the mouth is smiling.

Figures 4.4(d) and 4.5(d) show the shapes generated varying the third mode. It can be seen that this mode has captured mainly the oberture of the eyes varying from sleepy to normal.

4.3 Non-Linear Shape Modelling

Linear models have proved to be successful in capturing the variability present in some applications. However, there are some situations where they fail due the non-linearity of the relations between the landmark points. Consider for example the set of shapes in figure 4.6(a). After an initial linear decomposition the shapes generated by varying the first two PCs are shown in Figures 4.6(b) and 4.6(c). Figure 4.6(d) show the scattergram of these first two PCs. There is a clearly non-linear relationship between the aspect ratio of the external square and the size of the internal one. In addition, whether using a linear PDM, two modes are needed to explain the most of the variance, when one non-linear mode would be enough. Furthermore, the shapes generated by varying the linear PCs are not plausible with those on the training set, i.e. the model is not specific.

To overcome these problems Sozou et al. [41] proposed Polynomial Regression Point Distribution Models (PRPDM). It is based on a polynomial regression to capture the non-linearity of the shape variability.
Figure 4.4: Eigenvalues and first three variational modes of the model built from the shape representations of the example configurations.
Figure 4.5: Eigenvalues and first three variational modes of the model built from the size-and-shape representations of the example configurations
(a) Training set. Each shape is composed by eight points, four in the external square and four in the internal one.

(b) Shapes generated by varying the 1st mode from -3 to 3 s.d.

(c) Shapes generated by varying the 2nd mode from -3 to 3 s.d.

(d) Scattergram of the first two modes of variation

Figure 4.6: Example of non-linear relationships in shapes
4.3.1 Polynomial Regression Point Distribution Models

The basic idea is to allow polynomial behaviour as each shape parameter is varied, permitting more complex relationships between landmark points. This approach is motivated by noting that the eigen-analysis used to extract the modes of variation for a standard PDM can be conceptualized as a sequential process. From now on, we shall consider $x_i$ to be the vector representation of the $i$th configuration matrix, $x_i = \text{vec}(X_i)$.

1. For each training example $i$, initialize a vector of residuals $r_i$ to the deviation of that example from the mean: $r_i = x_i - \bar{x}$

2. Fit a straight line $u$ to the set of residuals \{r$_1$, ..., r$_n$\}, so as to minimize the sum of squares of distances from the straight line.

3. Compute the residual deviation from the line $u$ (i.e. remove the component of the $r_i$ along $u$) to give a new set of residuals: $r_i^{\text{new}} = r_i - (r_i \cdot u) \cdot u$

Steps 2 and 3 are repeated to find subsequent modes. We normally fit a sufficient number of modes for the remaining residuals to be small enough to be attributed to random noise (typically when the unexplained variance is between 5% and 1% of the original variance).

Note that, once a linear mode has been extracted, it may be possible to further reduce the residuals $r_i$ by fitting a polynomial along the direction of the mode. For example in Figure 4.6(d), the second PC could be modelled by a polynomial in the first PC. This leads to the idea of a polynomial regression model. The details of the procedure for building a PRPDM are as follows:

Algorithm: PRPDM

1. Compute initial variance of the data $\sigma_1^2$. Then, for each training example, initialise a vector of residuals $r_i$ to the vector of landmark positions $x_i$ (i.e. let $r_i = x_i$)

2. Let the current residual vector for each example $i$ be denoted by $r_i$, let the mean value of the $r_i$ be denoted by $\bar{r}$ and let the covariance matrix of the set of current residual vectors $\{r_1, ..., r_n\}$ be denoted by $S$:

$$S = \frac{1}{n} \sum_{i=1}^{n} (r_i - \bar{r})(r_i - \bar{r})^T$$

Find the eigen-vectors of $S$ denoted $|\phi_1| |\phi_2| ... |\phi_t|$. Define $t$ the true linear dimensionality of the set of residuals, as the number of eigenvalues needed to explain all of the variance in the residuals, apart from a tiny amount (which we set to $10^{-10}\sigma_1^2$) which may be considered as arising from rounding errors.
3. Transform the \( r_i \)s to a coordinate system with the axes lying along the eigenvectors, with \( t \) ordinates (corresponding to the \( t \) eigenvectors which explain all the variance other than that due to rounding errors). The new coordinates can then be labelled as:

\[
b_i = (b_{i1}, \ldots, b_{it})^T = \Phi^T (r_i - \bar{r}) \tag{4.10}
\]

where \( \Phi = (\phi_1 | \phi_2 | \ldots | \phi_t) \). Set \( l = 1 \).

4. Set the shape parameter for the mode being fitted to the first \( b \)-value, for each training example. Thus if the \( k \)th mode is being fitted, then \( b_{i1} \) is the \( k \)th shape parameter of the model for example \( i \).

5. Model the second and subsequent \( b \)-values as non-linear functions of the first:

\[
b_{ij}^* = f_j^l (b_{i1}) \quad \text{where } 2 \leq j \leq t \tag{4.11}
\]

These non-linear functions used in 4.11 are polynomials:

\[
f_j^l (b_{i1}) = a_{0j} + a_{1j} b_{i1} + a_{2j} b_{i1}^2 + \ldots + a_{mj} b_{i1}^m
\]

where \( m \) is the order of the polynomials. The coefficients \( a_{0j}, \ldots, a_{mj} \) (where \( 0 \leq t \leq m \) and \( 2 \leq j \leq t \)) are chosen so as to minimize the residual error \( E \) given by:

\[
E = \sum_{j=2}^t E_j
\]

where

\[
E_j = \sum_{i=1}^n (b_{ij}^* - b_{ij})^2
\]

The coefficients \( a_{0j} \) are obtained by partially differentiating \( E_j \) with respect to each \( a_{0j} \) and setting the derivative equal to zero. This gives \( m + 1 \) linear equations in \( m + 1 \) unknowns, which can be solved by standard methods.

6. Store the coefficients \( a_{0j}^l \) and the number of mode being fitted \( k \) as \( k_l \), in the actual iteration \( l \).

7. Compute the new residual vectors \( r_{i1}^{new} \) whose components are given by the difference between the data \( b \)-value \( b_{ij} \) and the modelled value \( b_{ij}^* \). Note that by definition the first \( b \)-value \( b_{i1} \) is the same as for the model and the data. We therefore ignore this component and set the first component of the new residual vector as the residual error in the second \( b \)-value. The
other components are similarly shifted down, and thus the new residuals will have \( t - 1 \) components given by:

\[
    r^\text{new}_{ij} = b_{i,j+1} - b^*_{i,j+1}
\]  

(4.12)

where \( 1 \leq j \leq t - 1 \)

8. Increment the value of \( l \) and repeat steps 2 to 8. Do it until the variance of the remaining residuals is under a desired threshold.

Once this procedure finishes one shall end with \( \bar{r} = \bar{x} \) the mean shape from the training set, \( \Phi \) a matrix of \( t \) eigenvectors; \( f^l_j \), \( p \) sets of polynomial coefficients; and \( k_l, p \) numbers of mode; being \( p \) the number of iterations needed to achieve the desired variance.

Given a new shape vector \( x_i \), the procedure that must be followed in order to get the non-linear parameters is:

**Algorithm: PRPDM projection**

Before we can get the non-linear parameterizations of a given shape, we must first compute the parameters of the PRPDM, following the algorithm above. After the algorithm we get \( k_l \), a vector indicating the number of the mode being fitted at each step, and \( f^l_j \) some sets of polynomials, being \( 1 \leq l \leq p \), and \( p \) the number of steps needed to achieve the desired variance when computing the PRPDM.

Then, given a shape instance \( x_i \) the nonlinear parameterization \( c \) is given by the following steps:

1. Get the linear projection \( b_i \) of the shape

\[
    b_i = \Phi^T (x_i - \bar{x}) ,
\]

where \( \Phi \) is the eigenvectors matrix obtained from step 3 of the algorithm to compute the PRPDM.

Set \( c \), the vector of non-linear parameterizations, initially to be a void vector. Set \( l = 1 \).

2. Get the vector of estimates \( b_i^* \) with the corresponding \( k_l \) and set of functions \( f^l_j \)

\[
    b^*_l j = f^l_j (bk_i),
\]

(4.13)

where \( k_l \) is the \( l \)th element of the vector \( k \), \( f^l_j \) is the polynomial used to estimate the \( j \)th mode from the \( k_l \)th mode.

Append the value of \( bk_i \) to \( c \). Note that \( b^*_l \) have one less dimension than \( b_i \).
3. Remove the parameter $b_{ik}$ from $b_i$, reducing in one its dimensions and thus, having the same length as $b_i^*$, and compute the residual.

$$b_i = b_i - b_i^*$$  \hspace{1cm} (4.14)

4. Increment $l$ by one and repeat steps 2 to 4 while there are more iterations to do ($l \leq p$).

The vector of non-linear parameterizations for shape $x_i$ is $c$.

Given a parameterization $c$ obtained by the above algorithm, one can get the original shape representation $x_i$ by following the procedure below:

Algorithm: PRPDM reconstruction

Before we can reconstruct the original shape from a given nonlinear parameterization $c$, we must first compute the parameters of the PRPDM, following the algorithm above. After the algorithm we get $k_l$, a vector indicating the number of the mode being fitted at each step, and $f_{lj}$ some sets of polynomials, being $1 \leq l \leq p$, and $p$ the number of steps needed to achieve the desired variance when computing the PRPDM.

Given a nonlinear parameterization $c$, one can get the original shape representation $x_i$ with the following steps:

1. Set $r_i$ to zero and $l$ to point the position of the last parameter in $c$.

2. Compute the vector of estimates

$$b_i^* = f_{lj}^j(c_l) + r_i$$  \hspace{1cm} (4.15)

where $c_l$ is the $l$th element of the vector $c$, and $f_{lj}^j$ is the polynomial used to estimate the $j$th mode from the $k_l$th mode.

3. Let the vector of residuals $r_i$ be the result of appending $c_l$ to their corresponding position in $b_i^*$ (indicated by the value of $k_l$).

4. Decrement $l$ by one, and repeat steps 2 to 4 until the first parameter of $c$ has been treated ($l = 1$).

5. Reconstruct the original shape

$$x_i = \Phi r_i + \bar{x},$$

where $\Phi$ is the eigenvectors matrix obtained from step 3 of the algorithm to compute the PRPDM.
Chapter 5

Modelling Shape Variability with Intermittently Present Landmarks

When modelling the shape of an object it is possible the samples that compose the class to be represented by a different number of landmark points (e.g., faces with or without moustache). The only approach we know that permits shapes with a non-constant number of points is by M. Rogers and J. Graham [37]. It is based on the classical PDM in Section 4.2.1, and combines data imputation (see Section 5.1) with a model of structural variation. It is called Structured Point Distribution Models (SPDM) and the particular example that led to their development is the study of electron microscope images of nerve capillaries.

SPDM has proven to be successful in capturing the patterns of presence and absence of the landmarks, however there are some situations that it fails to capture shape variability due to the non-linear nature of the landmarks variation. With the aim of extending the intermittently present data modelling to the non-linear case, we have defined a new model useful to deal with the non-linear variability as the PRPDM (section 4.3.1) and intermittently present data as in SPDM. For this reason we have called it Polynomial-Regression Structured Point Distribution Models (PR-SPDM). This new model is thought to be used in combination with a novel imputation scheme developed by us, called PR-PCA (Section 5.1.3). The aim is to impute the missing data according to their non-linear relationships, and after model these relationships as in PRPDM (section 4.3.1). We obtain good results in a synthetic example (see results in section 5.4).

In section 5.1 we present some imputation methods, in section 5.2 we describe the SPDM. In section 5.3 we describe our non-linear generalization PR-SPDM, in section 5.4 we present some experiments and results, in section 5.5 we do
some discussion, and in section 5.6 we present our conclusions and future work.

5.1 Imputation Methods

When modelling shapes with intermittently present landmarks we must do statistics with vectors of different lengths. Standard statistics do not permit non-rectangular datasets. Is for this reason that special methods are needed. In this Chapter we describe how to perform statistical analysis with vectors of different lengths. Specifically, we concentrate on methods to fill-in (impute) the missing-data so as to end up with a rectangular dataset.

Imputation methods consist on predicting the missing data values from the observed ones. Then the missing data are filled in (imputed) using these predicted values.

Suppose we have $Y_j$ (e.g height) missing and another variable $Y_k$ (e.g. weight) that is highly correlated with $Y_j$. It is tempting to predict the missing value of $Y_j$ from $Y_k$, and then to include the filled-in value in an analysis involving $Y_j$.

Moreover, when modelling shapes with intermittently present landmarks there is no real underlying value for the missing landmarks. However we need all the shapes that compound the class to be formed by the same amount of landmark points and therefore some imputation scheme is needed. As we shall see, the data values used for imputation play an important role in the resulting model and hence, imputing the missing-data according to the relationships present in the data is an important issue.

Next, we describe the following imputation methods: mean imputation, PCA imputation, and a novel method developed by us called PR-PCA imputation. The two latter approaches are iterative methods that impute the missing-data according to the first PCs.

5.1.1 Mean Imputation

The simplest method is to replace each missing value with the mean of the values that are present. Then for the examples with the $j$th variable missing $Y_{miss(j)}$:

$$Y_{miss(j),j} = \frac{1}{s_j} \sum_{Y_{obs(j),j}}$$

where $Y_{ij}$ is the $j$th variable of the $i$th sample, $s_j$ is the number of samples with the $j$th variable observed, and $Y_{obs(j)}$ is the set of samples with the $j$th variable observed.

5.1.2 PCA Imputation

Dear [15] proposed an imputation technique in which initial imputation with the means is then re-estimated using the first principal component of the imputed data.
data. The aim of this imputation technique is to impute values in such a way as to retain relationships found in the original data. Given $N$ observations of $k$ variables, where $x_i = (x_{i1}, \ldots, x_{ik})$ is the $i$th observation, the algorithm can be described by the following equations:

\[
\bar{x} = \frac{1}{N} \sum_{i=1}^{N} x_i, \quad (5.2)
\]

\[
S = \frac{N}{N} \sum_{i=1}^{N} (x_i - \bar{x})(x_i - \bar{x})^T \quad (5.3)
\]

\[
\Phi = [\phi_1 \ldots | \phi_k] = \text{eigenvectors} (S) \quad (5.4)
\]

\[
b_i = \phi_1^T x_i \quad (5.5)
\]

\[
\hat{x}_i = \bar{x} + \phi_1 b_i \quad (5.6)
\]

\[
x_{\text{miss}(j),j} = \hat{x}_{\text{miss}(j),j}, \quad 1 \leq j \leq k \quad (5.7)
\]

where $\text{eigenvectors} (S)$ is a function that computes the eigenvectors of the matrix $S$, and $\phi_1$ is the eigenvector associated to the largest eigenvalue.

We begin by initializing $x$, for which we use mean value imputation and cycle through Equations 5.2-5.7 until convergence. The procedure consists on modifying the missing data according to the first PC. After the update, the procedure is repeated with the results of the previous iteration.
Figure 5.1 illustrates the idea of PCA imputation. The fully observed examples are represented by crosses, while the examples with the missing vertical coordinate are represented by circles lying on the horizontal axis. The first PC is represented by a solid line while the dots represent the resulting positions of the missing-data examples after PCA imputation.

### 5.1.3 PR-PCA Imputation

This method has been developed by us with the aim of extending PCA imputation to the non-linear case. The idea is to impute the missing values according to the first non-linear PC. It follows the idea of PRPDM in Section 4.3.1 to extract the first non-linear PC, and so it is called PR-PCA imputation. Given $N$ observations of $k$ variables, where $x_i = (x_{i1}, \ldots, x_{ik})$ is the $i$th observation, the algorithm has the following steps:

1. Compute the mean vector $\bar{x}$ and eigenvectors matrix $\Phi$ with Equations 5.2-5.4.

2. Get the parameterizations $b_i$, the number of mode being fitted $k_1$ and the set of polynomials for the rest of the modes $f_{ij}^1$ using steps 3-5 of the algorithm to build a PRPDM in Section 4.3.1, with $r_i \equiv x_i$ and $\bar{r} \equiv \bar{x}$, $1 \leq i \leq N$.

3. Compute the vector of estimates $\hat{x}_i$ using steps 1-5 from the procedure to reconstruct the original representation given a PRPDM parameterization in Section 4.3.1. Use $c \equiv b_{ik_1}$, a length-1 vector, the eigenvectors matrix $\Phi$ and mean vector $\bar{x}$ from step 1. After these steps, $\hat{x}_i$ is the result of the reconstruction.

4. Update the missing variables with the reconstructed values:

   $$x_{\text{miss}(j),j} = \hat{x}_{\text{miss}(j),j}, \quad 1 \leq j \leq k$$

We begin by initializing $x$, for which we use mean value imputation and cycle through steps 1-4 until convergence. After the each update, the procedure is repeated with the results of the previous iteration. At each iteration the missing values are imputed according with the extracted polynomial PC.

Figure 5.2 illustrates the idea of PRPCA imputation. The fully observed examples are represented by crosses, while the examples with the missing vertical coordinate are represented by circles lying on the horizontal axis. The first PCs are represented by solid lines while the dots represent the resulting positions of the missing-data examples after imputation. When the data is nonlinearly correlated PRPCA succeeds in capturing the relationships as illustrated in Figure 5.2(a), while the linear regression obtained by the PCA imputation does not properly fit the data, as illustrated in Figure 5.2(b).
5.2 Structured Point Distribution Models

The aim of this model is to build a PDM that deals with intermittently present landmarks.

Let $x$ be an initial training set composed by $N$ configurations of $k$ landmarks in two dimensions.

$$x_i = [(x_{i1}, y_{i1}), (x_{i2}, y_{i2}), \ldots, (x_{ik}, y_{ik})].$$

Let each configuration be represented by the data vector

$$x_i = (x_{i1}, x_{i2}, x_{i3}, \ldots, x_{ik}, y_{i1}, y_{i2}, y_{i3}, \ldots, y_{ik})^T.$$

And let $\text{miss}(i)$ be the set of missing landmarks in the $i$th configuration. The procedure to build an SPDM is the following:

1. Replace each missing value by a placeholder such as NaN (a computational representation of Not a Number).

$$x_{i,\text{miss}(i)} = y_{i,\text{miss}(i)} = \text{NaN}, \quad 1 \leq i \leq N$$

2. Align the configurations using the landmarks that are available with either full or partial GPA of Section 3.5.

3. Impute the missing landmarks by some imputation scheme (see Section 5.1 for further details).

Then, an initial training shape $i$ with $\text{miss}(i) = \{2, 3\}$ becomes

$$\hat{x}_i = (x'_{i1}, \hat{x}_{i2}, \hat{x}_{i3}, \ldots, x'_{ik}, \hat{y}_{i1}, \hat{y}_{i2}, \hat{y}_{i3}, \ldots, \hat{y}_{ik})^T.$$
where primed elements are aligned and hat elements are imputed.

4. Perform a shifting and scaling in order to make the data \( \hat{x}' \) to lie between 0 and 1.

\[
\tilde{x}_i = \frac{x_i - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}}, \quad 1 \leq i \leq N, \tag{5.8}
\]

where \( x_{\text{max}} \) and \( x_{\text{min}} \) are the vectors with the maximum and minimum values of \( \hat{x}' \) for each dimension, respectively.

This shifting and scaling is to avoid problems associated with shape and structure being measured on different scales. It responds to the consideration of treating shape and structure as equally important.

5. Build a classical PDM as in Section 4.2.1, getting a parameterization \( b^d \) for the normalized training shapes

\[
\tilde{x} = \bar{x} + \Phi b^d, \tag{5.9}
\]

where \( \Phi \) are the eigenvectors of the covariance matrix of the training shapes coordinates in the tangent space, and \( \tilde{x} \) is the procrustes mean shape of Section 3.5.

While this gives us a model of shape variation, we have lost the structural information about which landmarks are present and which are not.

6. Augment the shape vector with an structure vector informing about the presence or absence of each landmark.

\[
x_s^i = (x_s^{i1}, x_s^{i2}, \ldots, x_s^{ik})^T,
\]

where \( x_{ij}^s = \{0, 1\} \) depending on whether the \( j \)th landmark of the \( i \)th shape \((x_{ij}, y_{ij})\) is present or not.

7. Apply PCA as in Section 4.1 on these structure vectors in order to reduce the redundancy, obtaining a reduced parameter vector \( b^s \) representing the structure vector

\[
x^s = \tilde{x}^s + Pb^s, \tag{5.10}
\]

where \( P \) are the eigenvectors of the covariance matrix of the structural data \( x^s \), and

\[
\tilde{x}^s = \frac{1}{N} \sum_{i=1}^{N} x_i^s \tag{5.11}
\]

is the mean structure vector of the \( N \) samples.
8. Build a combined model of shape and structure. i.e., for each training shape generate a concatenated vector

\[ b = \begin{pmatrix} b^d \\ b^s \end{pmatrix} = \begin{pmatrix} \Phi^T (\tilde{x} - \bar{x}) \\ P^T (x^s - \bar{x}^s) \end{pmatrix} \] (5.12)

9. Apply PCA again to obtain a combined model of shape and structure

\[ b = Qc \] (5.13)

where \( Q \) are the eigenvectors and \( c \) is a vector of structural shape parameters controlling both the position and presence or absence of the shape points. Since the shape and structural parameters have zero mean, \( c \) does too.

Note that the linear nature of the model allows to express the shape and its structural information directly as functions of \( c \)

\[ \tilde{x} = \bar{x} + \Phi Qdc , \quad x^s = \bar{x}^s + PQs c \] (5.14)

where

\[ Q = \begin{pmatrix} Q_s \\ Q_d \end{pmatrix} \] (5.15)

and the original shape is given by

\[ x = \bar{x} (x^{max} - x^{min}) + x^{min} \] (5.16)

An example shape can be synthesised for a given \( c \) by generating the shape from the vector \( x \) and removing those landmarks according to \( x^s \) and a given threshold.

The SPDM like the PDM is a generative model, i.e., given a parameter vector we can recreate the structure vector for a particular instance. The main drawback comes from the fact of representing a binary process (presence or absence) by a linear model. Thus, to recover binary parameters in the reconstructed structure vector, a threshold representing the probability of presence or absence, must be applied.

By the inclusion of the structure vector \( x^s \) we allow for arbitrary patterns of inclusion/exclusion.

### 5.3 Polynomial Regression Structured Point Distribution Models

The aim of this model is to build a PDM that deals with intermittently present landmarks and non-linearity in shapes.
Let $x$ be an initial training set composed by $N$ configurations of $k$ landmarks in two dimensions.

$$x_i = [(x_{i1}, y_{i1}), (x_{i2}, y_{i2}), \ldots, (x_{ik}, y_{ik})].$$

Let each configuration be represented by the data vector

$$x_i = (x_{i1}, x_{i2}, x_{i3}, \ldots, x_{ik}, y_{i1}, y_{i2}, y_{i3}, \ldots, y_{ik})^T.$$

And let miss$(i)$ be the set of missing landmarks in the $i$th configuration.

The procedure to build a PR-SPDM is the following:

1. Follow the steps 1-4 of the procedure to build an SPDM in Section 5.2. After these steps, we end up with an aligned set of shapes with the missing landmarks imputed, and this whole set normalized to lie between 0 and 1. We denote this set as $\tilde{x}$.

2. Build a classical PRPDM with the dataset $\tilde{x}$, as indicated by the algorithm in Section 4.3.1. Get the nonlinear parameterizations $b_{pr}$ for each shape $\tilde{x}$, as indicated in the same Section.

3. Augment the shape vector with an structure vector $x_s^i$ informing about the presence or absence of each landmark as in step 6 of the SPDM procedure in section 5.2.

4. Apply PCA to the structure vectors $x_s^i$ in order to reduce the redundancy, and get an eigenvectors matrix $P$ for the structure parameterization as in step 7 of the SPDM procedure in section 5.2.

5. Build a concatenated vector of shape and structural parameterizations

$$b = \left( \begin{array}{c} b_{pr} \\ b_s \end{array} \right)$$

(5.17)

6. Apply PCA again to obtain a combined model of shape and structure

$$b = Qc$$

(5.18)

where $Q$ are the eigenvectors and $c$ is a vector of structural shape parameters controlling both the position and presence or absence of the shape points. Since the shape and structural parameters have zero mean, $c$ does too.

Note that a shape and its structural information can be recreated for a given $c$.

$$b_{pr}^d = Q_{dc} \quad , \quad x^s = \tilde{x}^s + PQ_sc$$

(5.19)
where

\[ Q = \begin{pmatrix} Q_s \\ Q_d \end{pmatrix} \]  

(5.20)

and \( \tilde{x} \) is computed by applying the PRPDM reconstruction algorithm in Section 4.3.1 using \( c \equiv b_{pr}^d \).

Finally, the original shape is given by

\[ x = \tilde{x} \left( x_{\text{max}} - x_{\text{min}} \right) + x_{\text{min}} \]  

(5.21)

An example shape can be synthesised for a given \( c \) by generating the shape from the vector \( x \) and removing those landmarks according to \( x^s \) and a given threshold.

5.4 Experiments and Results

We have applied PDM, PRPDM, SPDM and PR-SPDM on the synthetic set of shapes of Figure 5.3.

While a vertice on the external square move along a straight line, the internal square appears, grows up, decreases, and finally disappears. There is a quadratic dependency between the aspect ratio of the external square and the area of the internal one, with a threshold that forces the internal square to disappear when its area is below 0.

As seen in Figure 5.3 the shapes at the extrema of the training set have only the external square while the rest of the figures have both the internal and the external one. Both the external and the internal square are composed by 16 landmark points. Thus, the figures with both squares are composed by 32 landmark points in two dimensions which gives vectors of length \( 32 \times 2 = 64 \). While the figures with only the external square are composed by 16 landmark points which gives vectors of length \( 16 \times 2 = 32 \). Thus, we have a dataset composed by vectors of length both 32 and 64.

5.4.1 PDM and PRPDM

Since both PDM and PRPDM need the shapes to be composed by the same number of points, we have used an imputation method (see Section 5.1) to fill the missing data in order to have the shapes with the same number of points. The two imputation methods used in the experiments are mean imputation and PR-PCA imputation.

The aim of this experiment is to see the advantages (if any) of using PRPDM instead of PDM in the example presented where non-linear correlations are present.

Figure 5.4 shows the scattergram and the two main modes of variation of PDM from the training set with mean imputation. In Figure 5.4(a) the first
Figure 5.3: Sombe objects of the training set. There are 16 points representing the figures with only the external square, and 32 the figures with both squares (there are 3 points along each line segment).
two PCs are superimposed, while Figures 5.4(b) and 5.4(c) show the shapes generated by varying the first and second PC respectively.

Figure 5.5 shows the scattergram and the first polynomial mode of variation of PRPDM from the training set with mean imputation. In Figure 5.5(a) the first polynomial PC is superimposed, while Figure 5.5(b) show the shapes generated by varying the first polynomial mode.

Figure 5.6 shows the scattergram and the first polynomial mode of variation of PRPDM from the training set with PR-PCA imputation. In Figure 5.6(a) the first polynomial PC is superimposed, while Figure 5.6(b) show the shapes generated by varying the first polynomial mode.
(a) Scattergram of the training with mean imputation. The first polynomial PC is superimposed.

(b) Shapes generated by varying the first polynomial PC from -1.6 to 1.6 s.d.

Figure 5.5: Scattergram and shapes generated by varying the first polynomial mode of the PRPDM on the training set with mean imputation.
(a) Scattergram of the training with PR-PCA imputation. The first polynomial PC is superimposed

(b) Shapes generated by varying the first polynomial PC from -1.6 to 1.6 s.d.

Figure 5.6: Scattergram and shapes generated by varying the first polynomial mode of the PRPDM on the training set with PR-PCA imputation
5.4.2 SPDM and PR-SPDM

We have applied both SPDM and PR-SPDM on the synthetic training shapes in figure 5.3.

Figure 5.7 shows the results of applying SPDM on the training set above. Both mean imputation and PCA imputation present nearly the same results in this case, so they are shown only once. The number of elements chosen in shape and structure parameterizations of equation 5.12 are 2 and 1 respectively. Therefore we will get at most, 3 modes of variation in the SPDM for this example. The threshold chosen for the reconstruction of the structural information of eq. 5.10 is 0.8.

Figure 5.8 shows the results of applying PR-SPDM with mean imputation on the training set above. The number of elements chosen for shape and structure parameterizations of equation 5.17 are both 1. Therefore we will get at most, 2 modes of variation in the PR-SPDM for this example. The threshold chosen for the reconstruction of the structural information is 0.8.

Figure 5.9 shows the results of applying PR-SPDM with PR-PCA imputation on the training set above. The number of elements chosen in shape and structure parameterizations of equation 5.17 are both 1. Therefore we will get at most, 2 modes of variation in the PR-SPDM for this example. The threshold chosen for the reconstruction of the structural information is 0.8.

Figure 5.11 shows the plots resulting from an evaluation experiment done to compare the representation ability of the different approaches. We perturbed the \((x, y)\) coordinates of the training shapes with an increasing amount of random noise, controlled by a noise coefficient. Fig 5.10 shows the resulting shapes after applying the minimum (no noise) (fig 5.10(a)) and the maximum (fig 5.10(b)) amount of noise on one of the shapes on the training set. We calculate the mean squared error between the reconstructed shapes and the original ones, for the whole training set, using each one of the models described. We are using shape and structure parameterizations of length 1 in both SPDM (eq. 5.12) and PR-SPDM (eq 5.17). The results are presented in two different plots corresponding to shapes composed by one and two parts.

5.5 Discussion

5.5.1 Data Imputation

Mean imputation is equivalent to putting an interior square of mean size. In the scattergram of Figure 5.4 can be seen that the projection points corresponding to the shapes with imputed interior square, lie on an horizontal straight line at the origin of the vertical axe. This is because while the PC corresponding to the horizontal direction accounts for the aspect ratio of the external square, the PC corresponding to the vertical direction accounts for the size of the internal square. Then the figures with imputed interior square lie on the vertical zero position (vertical origin) which means “mean size interior square”.

62
Figure 5.7: Results for SPDM with mean imputation. The number of parameters chosen for shape and structure parameterization are 2 and 1 respectively.
Figure 5.8: Results for PR-SPDM with mean imputation. The number of parameters chosen for shape and structure parameterization are 1 and 1 respectively.
Figure 5.9: Results for PR-SPDM with PR-PCA imputation. The number of parameters chosen for shape and structure parameterization are 1 and 1 respectively
Figure 5.10: Example shapes according to min and max values for the noise coefficient used in the experiment.

Figure 5.11: Mean squared errors for the reconstruction of 1 and 2 parts shapes of the whole training set for an increasing amount of noise.
On the other hand, PR-PCA imputation assigns an internal square of size according to the non-linear relationships captured by the polynomial axe. This can be seen in the scattergram of Figures 5.6 and 5.9.

Although there is no real underlying value for the missing-data when modelling shapes with intermittently present landmarks, the use of an imputation scheme that imputes the data so as to preserve the relationships brings more effectiveness to the models. It is demonstrated in the experiments. See Figures 5.5 and 5.6 to compare the adjustments obtained by the polynomial axes on the same dataset with either mean imputation or PR-PCA imputation.

5.5.2 Shape modelling

Results show how PRPDM fits better on the example presented due to the non-linear correlations on the data. Hence, PRPDM results in a more compact representation (i.e. fewer modes are needed) giving a more specific output (i.e. generating plausible elements, similar to those in the training set).

We have proven that PR-SPDM results in a better compactness in the example proposed, since there are needed only two modes to explain the three sources of variation found in the training set, say, presence/absence of the internal square, aspect ratio of the external square, and size of the internal square. Furthermore, PR-SPDM is more specific since it is more likely to generate valid shapes as seen in figures 5.8(c) and 5.9(c). On the other hand SPDM is not able to retain the relationships present in the training set as it encodes independently each source of variation in separate modes. It results in a model that do not represents the true variability and, therefore capable of generating invalid shapes as seen in figures 5.7(c) and 5.7(d).

On the other hand, PR-SPDM used in combination with PR-PCA imputation, has proven to best retain the underlying relationships of the present example, as it demonstrates the plots in the evaluation experiment.

As can be seen in the scattergrams (figs 5.7(a), 5.8(a) and 5.9(a)), the addition of the structural parameterizations into de model results in the stratification of the projection shapes into layers. i.e., all the projections that present the same pattern of presence/absence belong to the same level. In our example there are only two levels because there are only two patterns: with internal square and without it.

5.6 Conclusions and Future Work

We have demonstrated that PR-SPDM used in combination with PR-PCA imputation is the approach that best retains the underlying relationships of the synthetic example presented. We believe that this is portable to real examples e.g., medical images. We leave this study as future work.
Chapter 6

Publications derived from this Research Project

Bibliography


