Statistical Shape Analysis of Manufacturing Data

Enrique del Castillo

Department of Industrial and Manufacturing Engineering

The Pennsylvania State University, University Park, PA 16802, USA

November 9, 2009

Abstract

We show how Statistical Shape Analysis, a set of techniques used to model the shapes of biological and other kind of objects in the natural sciences, can be used also to model the geometric shape of a manufactured part. We review Procrustes-based methods, and emphasize possible solutions to the basic problem of having corresponding, or matching, labels in the measured "landmarks", the locations of the measured points on each part acquired with a CMM or similar instrument.

Keywords: Procrustes Methods, Point Matching, Correspondence Problem, Landmark Matching, Geometric Specifications, Profile data.

1 Introduction

In Statistical Shape Analysis (SSA) the *shape* of an object is defined as all the information of the object that is invariant with respect to similarity transformations on the Euclidean space (rotations, translations, and dilations or changes of scale). The goal of SSA is to analyze the shapes of objects in the presence of random error.

Analysis of shapes in manufacturing is critical because geometrical tolerances (specifications) of roundness, flatness, cylindricity, etc., need to be inspected, controlled, or optimized based on a cloud of 2 or 3 dimensional measurements taken on the machined surfaces of the part. These tasks are even more complex if the part geometry has a "free form", i.e., there is no standard geometrical construction that can represent the shape, a situation common in advanced manufacturing applications such as in the aerospace sector.

Over the last 20 years, statistical shape analysis techniques have been developed and applied in many areas of the natural sciences where interest is in characterizing differences between and variability within shapes, e.g., Biology, Paleontology and Geology. A considerable intersection of ideas exist also with image and Pattern Recognition in Computer Science. In particular, SSA is known as *Geometric Morphometrics* in Biology, and the type of techniques developed in the past two decades has been called the "Morphometrics revolution" by some authors [1] given the success SSA had over previous techniques used to analyzed shapes. For more on the history and foundations of SSA we refer readers new to this field to the excellent book by Dryden and Mardia [13].

Our interest on shape analysis stems in part from the recent interest on "profile analysis" in the field of statistical process control (SPC) [20, 8, 34] (although we do *not* discuss SPC based on shape analysis in this chapter, this is certainly another potential area of research where SSA ideas can be used). In profile-based SPC, a parametric model is sought that describes the form that the response follows with respect to some variable of interest (in essence, one performs functional data analysis). The parameters of this model are fitted based on process data and then multivariate SPC methods are applied to the estimated parameters.

By working with the shape directly, SSA techniques avoid the parametric model definition step, allow complicated shapes to be studied, and simplify the presentation of results. In SSA, one works with the whole shape of the object, so the geometry is not "thrown away" [13].

The remainder of the paper is organized as follows. Section 2, describes methods to solve the landmark matching problem, which occurs when two objects have point labels that do not correspond to each other. In section 3, we review the main ideas on SSA based on the so-called Procrustes method. In this section, the notions of shape space, the generalized Procrustes algorithm, and tangent space coordinates are discussed. The paper concludes with a discussion of other shape analysis techniques, including areas of further work.

2 The Matching Landmarks Problem

In most of SSA, the main goal is statistical inference with shapes, in particularly, to test if two or more objects have an equal shape or not, or to determine directions where most of the variability of a shape occurs. Some other authors' main interest (for example, in Biology) is to describe how shapes of objects (e.g., species of animals) change with time. In our case, the main goal is to study the shape of manufactured parts.

The techniques considered herein are based on shape data obtained by measuring the parts at specific *landmarks*, points of special interest or unique characteristics. In order to be amenable to data analysis, landmarks should refer to homologous points (points of correspondence) that match between objects. A landmark is given by the 2 or 3 dimensional Cartesian coordinates of a point on the object surface and a given label for the point, usually a sequential number $1, 2, ..., k$ which corresponds from object to object. Assignment of landmarks to objects is in itself an important problem; in some areas such as in Archeology or Biology specific points of the objects are of interest and this assignment is done manually. In manufacturing, considerable amounts of data can be acquired with a coordinate measuring machine (CMM) or through digital images of the objects. There is no guarantee in practice, however, that the measurements acquired will correspond to each other between parts. Homologous landmarks have the same label, hence we call the case of complete homologous landmarks the *labeled* case.

All the SSA methods considered in later sections of this chapter require labeled landmark data. Similar parts measured with a CMM not always contain corresponding or labeled landmarks. This can be due to the difficulty in orienting the part when mounting it on the CMM. If the orientation is different between parts, the CMM measurements will not correspond to each other, since they will have different labels. Therefore, one first important problem that needs to be addressed is how to "match" the landmarks between 2 or more shapes so that we obtain corresponding shape data. This problem has received attention in the Pattern Recognition literature in recent years, where it is called the point matching or shape matching problem. The work by Ranjaragan and co-workers [14, 7] is based on solving a highly nonlinear optimization problem where the objective is to minimize the sum of the Euclidean distance between points $\{i\}$ in shape 1 and the transformed points $\{j\}$ in shape 2. The rationale for this approach is that matching would be relatively easier if the objects would be oriented similarly, and have similar location and scale (similarity transformations). Jointly determining the matching correspondences and the transformation necessary for "registering" object 2 to object 1 results in a hard optimization problem.

A completely different approach is that of Belongie et al. [3], who propose an efficient method for matching two 2-dimensional shapes, although they left undefined some implementation details, as we will see below. Their method separates the matching landmarks problem from the problem of registering the objects, that is, their matching method is in principle invariant with respect to location, scaling and orientation of the two parts. The main idea is to measure the amount of data in the neighborhood of each point of each shape (given by the frequency of points in its neighborhood) and use these measures as costs to be minimized in a classical weighted matching problem, solvable via Linear Programming. For a point i in a shape, Belongie et al. propose to compute a 2-dimensional histogram where the number of points nearby are counted. If r is the Euclidean distance between two points of the shape, the 2-dimensional histogram extends along $log\ r$ and θ, measuring the distance and direction where the nearby points are located. The histogram bins are selected such that they are of constant width in $(log\ r, \theta)$, giving more importance in this way to closer points. Let $h_i(l, s)$ be the observed frequency of nearby points in cell (l, s), of the histogram, where $l = 1, ..., L, s = 1, ..., S$. The

2-dimensional histogram formed by the frequencies $h_i(\cdot,\cdot)$ is called the "context" of point i by these authors. The idea then is to match those points between two different shapes that have the most similar "contexts". For this purpose, define the cost of matching point i in part 1 and point j in part 2 to be:

$$C_{ij} = \sum_{l=1}^{L}\sum_{s=1}^{S} \frac{[h_i(l,s) - h_j(l,s)]^2}{h_i(l,s) + h_j(l,s)}, \quad i = 1, 2, ..., k, j = 1, 2, ..., k \tag{1}$$

which is the classical χ^2 statistic (with $L \cdot S - 1$ degrees of freedom) used to test for the difference between two distributions. Note that $C_{ij} \neq C_{ji}$. Let $B = (U, V, E)$ be a graph with two disjoint sets of points (U and V), i.e., a bipartite graph, and a set of edges (E, to be decided) joining a point in U with a point in V (the "matching" set). Define the decision variables $X_{ij} = 1$ if the edge joining points $v_i \in V$ and $u_j \in U$ is included in the matching, and $X_{ij} = 0$ if otherwise [30]. Belongie et al. [3] propose to solve the landmark matching or labeling problem by solving the following weighted matching problem (in our notation):

$$\min \textstyle\sum_{i=1}^{k} \sum_{j=1}^{k} C_{ij} X_{ij} \tag{2}$$

subject to:

$$\begin{gathered} \textstyle\sum_{j=1}^{k} X_{ij} = 1, \quad i = 1, 2, ..., k \\ \textstyle\sum_{i=1}^{k} X_{ij} = 1, \quad j = 1, 2, ..., k \\ X_{ij} \geq 0, \quad i = 1, ..., k; j = 1, ..., k. \end{gathered}$$

The problem is then one of *linear* programming (LP), for which, as it is well-known, there exist efficient algorithms. Note that the formulation does not include the constraints $X_{ij} \in \{0, 1\}$, which turn out to be redundant (the LP solution is always binary) so the problem is *not* an integer programming problem, which would imply a considerable harder optimization problem. The matrix formulation of the problem is based on defining the $k^2 \times 1$ vector of decision variables

$$\boldsymbol{x}' = (X_{11}, X_{12}, ..., X_{1k}, X_{21}, X_{22}, ..., X_{2k}, ..., X_{k1}, X_{k2}, ..., X_{kk})$$

and defining the $2k \times k^2$ matrix of constraint coefficients:

$$\boldsymbol{A} = \left[\begin{array}{ccccccccccccc}
1 & 1 & \dots & 1 & & & & & & & & & \\
 & & & & 1 & 1 & \dots & 1 & & & & & \\
 & & & & & & & & \ddots & & & & \\
 & & & & & & & & & 1 & 1 & \dots & 1 \\
1 & & & & 1 & & & & \dots & 1 & & & \\
 & 1 & & & & 1 & & & \dots & & 1 & & \\
 & & \ddots & & & & \ddots & & \dots & & & \ddots & \\
 & & & 1 & & & & 1 & \dots & & & & 1
\end{array}\right]$$

14	41	21	25	9	39
21	42	22	19	15	39
29	42	25	22	21	40
35	37	9	39	25	36
32	33	21	27	23	31
26	30	21	40	21	27
16	26	15	39	19	25
25	26	8	17	21	25
29	24	25	36	23	24
33	20	15	17	25	22
30	16	19	25	22	19
23	11	23	31	15	17
16	12	23	24	8	17

Table 1: Input matrices for the "digit 3's" problem (first 4 columns). Last 2 columns is the output, sorted matrix.

where all empty spaces are zeroes. If matrix $\boldsymbol{C} = \{C_{ij}\}$ is put into vector form as follows

$$\boldsymbol{c}' = (C_{11}, C_{12}, ..., C_{1k}, C_{21}, C_{22}, ..., C_{2k}, ..., C_{k1}, C_{k2}, ..., C_{kk})$$

then the formulation is simply

$$\min \boldsymbol{c}'\boldsymbol{x}$$

$$\text{subject to:}$$

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$$

$$\boldsymbol{x} \geq \boldsymbol{0}$$

where $\boldsymbol{b}$ is a $2k \times 1$ vector of ones. The property that assures a $\{0, 1\}$ solution is that matrix $\boldsymbol{A}$ is a Totally Unimodal Matrix (TUM) ([30], Theorem 13.3). A matrix is TUM if 1) it has zeroes except in 2 locations per column, where it has ones; and 2) the rows can be grouped in two sets such that the ones in each column belong to different sets. These two properties hold for matrix $\boldsymbol{A}$.

An important implementation detail is how to scale the distances. We suggest to define $r_{ij} = d_{ij}/\max(d_{ij})$ where d_{ij} is the Euclidean distance between points i and j in the figure and the maximum is measured over all distances between any two points (landmarks). Therefore, $\max(r_{ij}) = 1$.

Example. Landmark Matching. Suppose we have the two shapes shown in Figure 1. These are two handwritten digit 3's, each with 13 landmarks. Suppose the landmarks are labeled as shown in the figure and in the first 4 columns of Table 1. We will keep the labels of shape 1 constant and will try to match the labels of the second shape to those of the first. The cost matrix $\boldsymbol{C}$ is shown on Table 2. This was obtained using a 2-dimensional histogram at each point of each shape where $L = 10$ bins where used for log r (logarithm of Euclidean distances) and $S = 9$ bins were used for θ. Specifically, the bin edges where set at $[0.0, \exp(-4), \exp(-3.5), \exp(-3.0), \exp(-2.5), \exp(-2.0), \exp(-1.5), \exp(-1), \exp(-0.5), 1]$ and $[-\pi, -3*\pi/4, -\pi/2, -\pi/4, 0.0, \pi/4, \pi/2, 3*\pi/4, \pi]$ (this is a higher resolution histogram that used by Belongie et al. [3]; we found the results vary considerably with the resolution of the histogram, given by the number of bins. Intuitively, the number of bins should be an increasing function of k, the number of landmark points). The costs C_{ij} were computed by excluding those cells in the histograms that would lead to a zero denominator. The structure of the $\boldsymbol{A}$ matrix, a 26×169 matrix, is shown diagrammatically in Figure 2. Solving the resulting LP problem (we used the `linprog` routine in MATLAB), the optimal solution leads to the correspondences shown in Figure 3. The re-ordered landmark matrix for the second figure is shown at the right of Table 1. ■

We point out how the two figures in the example did not have the same scale. We would like for a matching algorithm to work even if the figures are not equally oriented (i.e. to be rotation invariant; the method is already location and scale invariant). For figures with different orientation, a simple solution, which we used in the previous example, is to compute the angles θ in the histogram with respect to the line defined by the two closest points to the point in question (this is a similar procedure as suggested in [3], who suggest to use the “tangent” line to each point as the axis of reference).

An apparently open problem in the literature is how to solve similar matching problems when there are n shapes, not only two. A first attempt to such problem may involve matching shapes (1,2) obtaining 2’ (the re-labeled object 2), then match (2’,3), (3’,4),..., (n-1’,n) and then repeat matching (n’,1),(1’,2’), etc., until convergence. It is unknown how effective such approach is, and whether or not convergence is guaranteed.

3 A review of some statistical shape analysis concepts and techniques

There is a very large body of literature on SSA techniques. Only its main precepts and techniques are presented here. For a more thorough presentation of SSA we refer readers to references [13, 15] for developments up to 1998 and for more recent developments we refer to [1, 25, 24, 17].

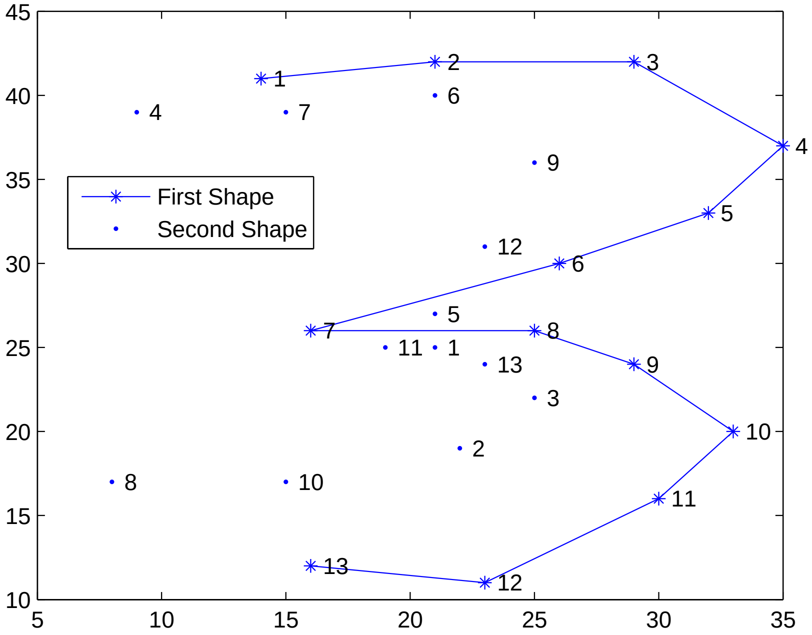

Figure 1: Two handwritten digit “3”s, each with $k = 13$ landmarks. The labels of the landmarks of the second digit were shuffled and do not correspond to those of the first digit 3.

The mainstream of SSA that followed the “revolution” in morphometrics is based on two main steps: first, the objects under consideration are registered or superimposed with respect to each other in order to filter out rotation, translation and isometric scaling (dilation) effects. This is done because the objects may have different orientations on the Euclidean space or have different locations or sizes, and therefore their shapes cannot be initially compared. The main technique for this task is the Generalized Procrustes Algorithm (GPA). An underlying assumption of GPA is that landmarks refer to homologous or corresponding points in each object. Since this is not always the case in CMM data, the landmark matching problem discussed in the previous section must be solved first before attempting the registration. Matching before registering seems to be a simpler and better strategy than trying to jointly match the landmarks and register the objects, as attempted in references [14, 7].

Once objects are registered, multivariate statistical methods of inference can be performed on the projections of the shapes on the space tangent to the mean shape. These two steps are explained below. We first give some geometrical notions necessary to understand the algorithms.

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	7.0	8.3	7.8	**4.7**	9.3	6.8	4.4	11.5	10.2	9.5	10.1	10.2	7.3
2	7.7	7.0	9.5	6.8	9.2	5.5	**2.0**	10.4	8.0	6.7	10.1	9.2	8.0
3	6.2	3.5	6.3	5.7	8.2	**1.6**	5.9	6.0	6.7	2.7	10.7	7.0	6.8
4	9.6	7.7	8.8	9.0	6.3	7.5	8.2	7.0	**2.6**	5.3	9.0	3.8	8.5
5	6.3	6.1	8.2	10.7	7.0	7.2	7.7	8.8	5.6	5.8	8.8	**3.5**	8.7
6	8.5	9.3	12.5	10.7	**6.5**	7.8	4.8	10.1	7.2	6.9	11.0	9.2	9.7
7	7.8	8.0	10.0	9.0	7.8	6.5	3.2	6.3	7.3	5.3	**9.0**	7.5	8.5
8	**5.0**	8.5	9.2	11.7	5.7	6.6	7.7	11.7	7.5	6.8	10.2	7.2	9.4
9	3.0	7.1	6.7	9.8	6.2	4.7	7.7	11.0	6.8	6.7	9.5	7.0	**5.2**
10	5.2	6.3	**4.5**	7.5	8.3	4.8	5.8	10.6	7.6	7.7	9.7	8.2	4.3
11	5.8	**1.3**	4.5	5.8	6.3	3.2	4.2	8.9	7.0	5.4	11.0	5.5	5.7
12	5.3	5.3	8.1	6.1	8.6	2.8	4.7	6.7	4.6	**2.0**	12.5	9.5	6.1
13	6.3	6.9	7.7	7.7	10.2	3.1	8.9	**4.5**	5.1	3.3	12.0	9.4	6.2

Table 2: The Cost matrix $\boldsymbol{C}$ for the two digit 3s problem. This is a $k \times k = 13 \times 13$ non-symmetric matrix. Bold numbers correspond to costs for the optimal matching solution.

3.1 Preshape and shape space

Let $\boldsymbol{X}$ be a $k \times m$ matrix containing the k landmarks (coordinate pairs or triples) of an object in m (2 or 3) dimensions. $\boldsymbol{X}$ is sometimes called a *configuration matrix*, which we could also refer to as a "profile matrix", following manufacturing practice for the case of 2D closed contours [2]. With this notation, the shape of a configuration $\boldsymbol{X}$ is obtained, first, by removing location and scale effects by computing the so-called *pre-shape* $\boldsymbol{Z}$:

$$\boldsymbol{Z} = \frac{\boldsymbol{H}\boldsymbol{X}}{||\boldsymbol{H}\boldsymbol{X}||} \tag{3}$$

where $\boldsymbol{H}$ is a $(k-1) \times k$ Helmert submatrix [13] and $|| \cdot ||$ denotes the Frobenius norm of a matrix (i.e., $||\boldsymbol{A}|| = \sqrt{\sum_i \sum_j |a_{ij}|^2}$). If we define $h_j = -[j(j+1)]^{-1/2}$, then $\boldsymbol{H}$ is a matrix whose jth row is: $(\underbrace{h_j, h_j, ..., h_j}_{j\text{ times}}, -jh_j, \underbrace{0, ..., 0}_{k-j-1\text{ times}})$ for $j = 1, ...k-1$. Note that $\boldsymbol{H}\boldsymbol{H}' = \boldsymbol{I}_{k-1}$ and that the rows of $\boldsymbol{H}$ are contrasts. Alternatively, one could start with the *centered preshapes*, defined by $\boldsymbol{Z}_c = \boldsymbol{H}'\boldsymbol{Z}$ (these are $k \times m$ matrices).

Transformation (3) removes location effects via the numerator, and re-scales the configurations to unit length via the denominator. Since we have not removed rotations from $\boldsymbol{Z}$ it is not yet the shape of $\boldsymbol{X}$, hence the name preshape. The centered preshapes are equivalent to centering each coordinate of each configuration by its centroid and dividing each by its norm.

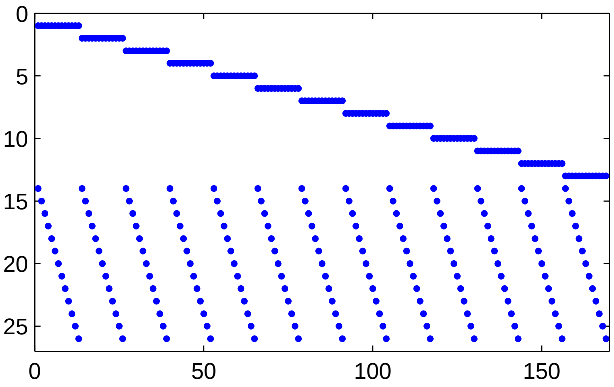

Figure 2: The $\boldsymbol{A}$ matrix in the weighted matching linear programming formulation applied to the 2 digit 3's matching problem (where $k = 13$). Dots indicate ones, empty spaces indicate zeroes. This is a $2k \times k^2 = 26 \times 169$ matrix.

The *shape* of configuration $\boldsymbol{X}$, denoted $[\boldsymbol{X}]$, is defined as the geometrical information that is invariant to similarity transformations. Once location and scale effects are filtered as above, the shape is then defined as:

$$[\boldsymbol{X}] = \{\boldsymbol{Z}\boldsymbol{\Gamma} : \boldsymbol{\Gamma} \in SO(m)\} \tag{4}$$

where $\boldsymbol{Z}$ is the preshape of $\boldsymbol{X}$ and $\boldsymbol{\Gamma}$ is a rotation matrix (i.e., a matrix such that $\boldsymbol{\Gamma}'\boldsymbol{\Gamma} = \boldsymbol{\Gamma}\boldsymbol{\Gamma}' = \boldsymbol{I}_m$ with $\det(\boldsymbol{\Gamma}) = +1$) and $SO(m)$ is the space of all $m \times m$ rotation matrices, the special orthogonal group. Multiplication by a suitable matrix $\boldsymbol{\Gamma}$ reorients (rotates) the object. Note that a shape is therefore defined as a set.

The following geometrical interpretation of these transformations is due to Kendall [21, 22]. Given that preshapes are scaled and centered objects, they can be represented by vectors on a sphere of dimension $(k-1)m$, because the numerator in (3) removes m degrees of freedom for location parameters and the denominator removes one additional degree of freedom for the change of scale. The preshapes, having unit length, are therefore on the surface of this (hyperspherical) space, which has $(k-1)m-1$ dimensions by virtue of being on the surface of a unit sphere. As one rotates a pre-shape $\boldsymbol{Z}$ via (4), the vectors $\boldsymbol{Z}\boldsymbol{\Gamma}$ describe an *orbit*, in effect, a geodesic, on the preshape space. All these vectors correspond to the same shape, since by definition the shape of an object is invariant to rotations. Thus, the orbits (also called *fibres*) of the preshape space are mapped one to one into single points in the *shape space*, the space where shapes reside. Two objects have the same shape if and only if their preshapes lie on the same

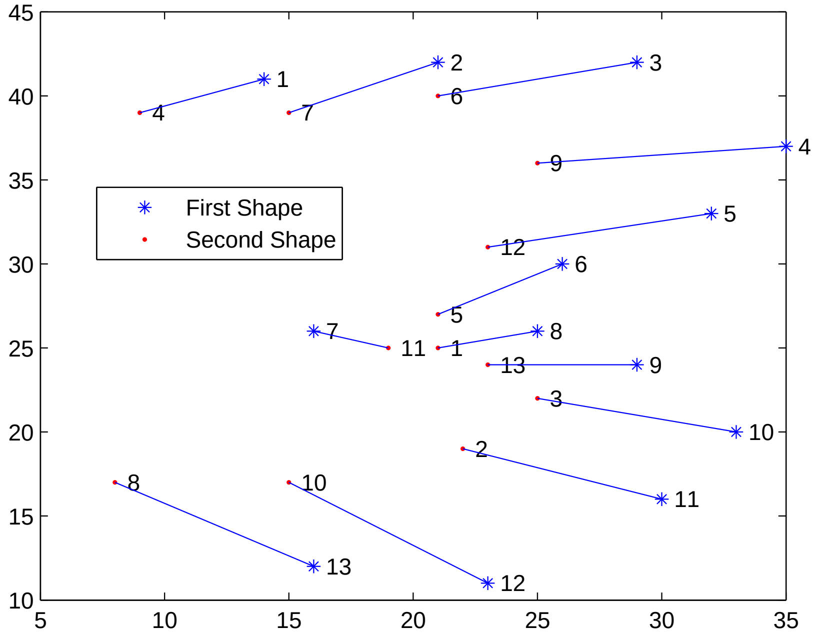

Figure 3: Optimal solution to the linear programming matching problem, digit 3 problem. Compare to Figure 1.

fibre. Fibres do not overlap. The shape space, the space of all possible shapes, has dimension $M = (k-1)m - 1 - m(m-1)/2$ since in addition to losing location and dilation degrees of freedom we also lose $m(m-1)/2$ degrees of freedom in the specification of the (symmetric) $m \times m$ rotation matrix $\mathbf{\Gamma}$.

Example. Preshape space and shape space. In order to explain these ideas, consider one of the simplest possible cases, where we have 2 lines in $\mathbb{R}^2$. Thus, we have that $m = 2$ and $k = 2$, where the obvious landmarks are the endpoints of the lines. After centering and scaling the two lines using (3), one obtains the preshapes with matrices $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$. Since the original objects evidently have the same shape (that of a line in Euclidean space) these two preshapes lie on the same fibre or orbit, generated as the preshapes are rotated using (4). The shape space is of dimension $(k-1)m - 1 = 1$, namely, the circumference of a circle. As the preshapes rotate (they can rotate clockwise or counterclockwise) they will eventually coincide, which corresponds to the centered and scaled lines coinciding. Finally, since there is a single shape, the shape space is the simplest possible, namely, a single point (dimension is $M = (k-1)m - 1 - m(m-1)/2 = 0$). ■

In general, the shape space will also be a spherical, nonlinear space, of reduced dimension than the preshape space.

3.2 Generalized Procrustes Algorithm

Two preshapes $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$ lying on different fibres correspond to two objects with different shapes. A measure of the similarity between two shapes is the shortest distance between the fibres, the *Procrustes distance* $\rho(\boldsymbol{X}_1, \boldsymbol{X}_2)$. This corresponds to the distance along the surface of the preshape space and is therefore a distance along a geodesic. Alternatively, two measures of distance over a linear space are the “partial procrustes distance”, given by

$$d_p(\boldsymbol{X}_1, \boldsymbol{X}_2) = \min_{\boldsymbol{\Gamma} \in SO(m)} ||\boldsymbol{Z}_2 - \boldsymbol{Z}_1 \boldsymbol{\Gamma}|| \tag{5}$$

and the “full procrustes distance”, where the minimization is also done over a scale parameter:

$$d_F(\boldsymbol{X}_1, \boldsymbol{X}_2) = \min_{\boldsymbol{\Gamma} \in SO(m), \beta \in \mathbb{R}} ||\boldsymbol{Z}_2 - \beta \boldsymbol{Z}_1 \boldsymbol{\Gamma}|| \tag{6}$$

Geometrically, $d_p(\boldsymbol{X}_1, \boldsymbol{X}_2)$ is the secant between $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$ in preshape space, and $d_F(\boldsymbol{X}_1, \boldsymbol{X}_2)$ is the distance along the tangent at either one of the preshapes (see Figure 4). As it can be seen, for objects with similar shapes, $\rho \approx d_F \approx d_p$.

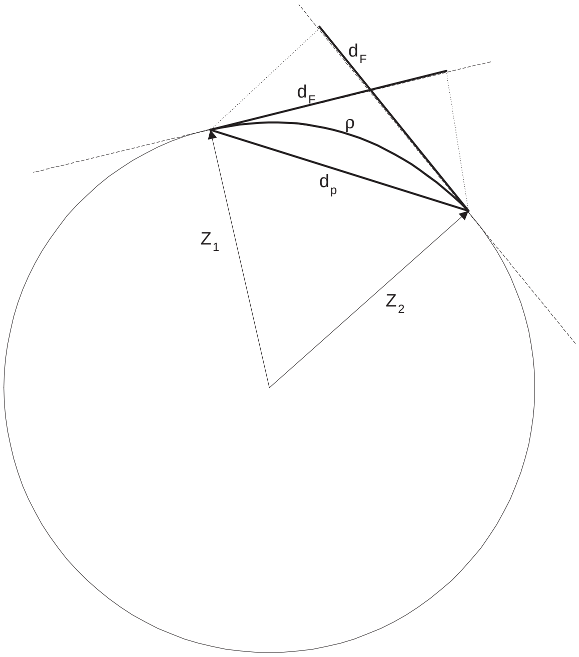

Figure 4: Distances between two shapes in preshape space. ρ is the Procrustes distance (along a geodesic), d_F is the Full Procrustes distance (along a tangent), and d_p is the Partial Procrustes distance (along the secant). The preshapes have $||\boldsymbol{Z}_i|| = 1$.

For a collection of n registered configurations or profiles, the *Generalized Procrustes Algorithm* registers or superimposes all the n objects by finding scaling factors $\beta_i \in \mathbb{R}$, rotation matrices $\mathbf{\Gamma}_i \in SO(m)$ and m dimensional translation vectors $\boldsymbol{\gamma}_i, i = 1, ..., n$, such that they minimize the sum of squared Full Procrustes Distances between all objects:

$$\begin{aligned} G(\boldsymbol{X}_1, \boldsymbol{X}_2, .., \boldsymbol{X}_n) &= \min_{\beta_i, \mathbf{\Gamma}_i, \boldsymbol{\gamma}_i} \frac{1}{n} \sum_{i=1}^{n} \sum_{j=i+1}^{n} ||\beta_i \boldsymbol{X}_i \mathbf{\Gamma}_i + \mathbf{1}_k \boldsymbol{\gamma}_i' - (\beta_j \boldsymbol{X}_j \mathbf{\Gamma}_j + \mathbf{1}_k \boldsymbol{\gamma}_j')||^2 \qquad (7) \\ &= \frac{1}{n} \sum_{i=1}^{n} \sum_{j=i+1}^{n} d_F^2(\boldsymbol{X}_i, \boldsymbol{X}_j) \end{aligned}$$

where $\mathbf{1}_k$ is a vector of k ones. The resulting registered configurations are called the *Full Procrustes Fits*, defined as

$$\boldsymbol{X}_i^p = \widehat{\beta}_i \boldsymbol{X}_i \widehat{\mathbf{\Gamma}}_i + \mathbf{1}_k \widehat{\gamma}_i', \;\; i = 1, ..., n. \qquad (8)$$

The mean shape of the n objects is simply the average of the n configurations, namely, $\widehat{\boldsymbol{\mu}} = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{X}_i^p$.

The minimization (7) needs to be subjected to a constraint that limits the scaling done, otherwise the optimal value of G will be zero. One such restriction is to use a constraint on the size of the mean shape: $S(\widehat{\boldsymbol{\mu}}) = 1$ where the size of any configuration $\boldsymbol{X}$ is defined as $S(\boldsymbol{X}) = \sqrt{\sum_{i=1}^{k} \sum_{j=1}^{m} (X_{ij} - \overline{X}_j)^2} = ||\boldsymbol{C}\boldsymbol{X}||$, where $\boldsymbol{C} = \boldsymbol{I}_k - k^{-1} 1_k 1_k'$, $\overline{X}_j = \frac{1}{n} \sum_{i=1}^{k} X_{ij}$ and X_{ij} is the jth coordinate of the ith point in the configuration. Another common constraint, used in what follows, is to make the average of the squared sizes of the registered configurations $\boldsymbol{X}_i^p$ given by (8) equal to the average of the squared sizes of the original objects:

$$\frac{1}{n} \sum_{i=1}^{n} S^2(\boldsymbol{X}_i^p) = \frac{1}{n} \sum_{i=1}^{n} S^2(\boldsymbol{X}_i). \qquad (9)$$

The Generalized Procrustes Algorithm, as developed by Gower [16] and Ten Berge [33] proceeds as follows to solve (7) subject to (9):

1. Center (but do not scale) the configurations $\boldsymbol{X}_1, ..., \boldsymbol{X}_n$ by initially defining

   $$\boldsymbol{X}_i^p = \boldsymbol{H}\boldsymbol{X}_i, \;\; i = 1, ..., n$$

   (alternatively, we can define $\boldsymbol{H}'\boldsymbol{H}\boldsymbol{X}_i = \boldsymbol{C}\boldsymbol{X}_i = \boldsymbol{X}_i^p$ and the resulting matrices will be $k \times m$; note that $\boldsymbol{X}_i^p$ as defined above is instead $(k-1) \times m$)

2. Let $\overline{\boldsymbol{X}}_{(i)} = \frac{1}{n-1} \sum_{j \neq i} \boldsymbol{X}_j^p, \;\; i = 1, ..., n$. These are the "jacknifed" average shapes excluding object i.

3. Do a Procrustes fit (rotation only) of the current $\boldsymbol{X}_i^p$'s on to $\overline{\boldsymbol{X}}_{(i)}$. This yields rotation matrices $\widehat{\mathbf{\Gamma}}_i$ from which we let

   $$\boldsymbol{X}_i^p \leftarrow \widehat{\mathbf{\Gamma}}_i \boldsymbol{X}_i^p, \;\; i = 1, ...n.$$

   We repeat steps 2 and 3 for all i.

4. Compute the $n \times n$ correlation matrix $\mathbf{\Phi} = corr(\boldsymbol{X}_v)$ where

$$\boldsymbol{X}_v = [\text{vec}(\boldsymbol{X}_1^p)\text{vec}(\boldsymbol{X}_2^p)...\text{vec}(\boldsymbol{X}_n^p)].$$

   where vec($\boldsymbol{X}$) returns a vector in which we stack the columns of matrix $\boldsymbol{X}$ one of top of each other. Note we stack all the m dimensions together.

5. Let $\boldsymbol{\phi} = (\phi_1, ..., \phi_n)'$ be the eigenvector of $\mathbf{\Phi}$ corresponding to its largest eigenvalue. Then set

$$\widehat{\beta}_i = \sqrt{\frac{\sum_{j=1}^n ||\boldsymbol{X}_j^p||^2}{||\boldsymbol{X}_i^p||^2}}\phi_i, \quad i = 1, ..., n$$

   and let $\boldsymbol{X}_i^p \leftarrow \widehat{\beta}_i \boldsymbol{X}_i^p$. The algorithm repeats steps 2 to 5 until convergence.

The algorithm is guaranteed to converge (in the sense that the fitted $\boldsymbol{X}_i^p$ cease to vary as i increases), usually in just a few iterations [33]. The exact solution to the Procrustes registration problem between two objects $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ required in step 3, implies finding $\mathbf{\Gamma} \in SO(m)$ that minimizes $d_p(\boldsymbol{X}_1, \boldsymbol{X}_2)$ (see 5) for $\boldsymbol{X}_1 = \boldsymbol{X}_i^p$ and $\boldsymbol{X}_2 = \overline{\boldsymbol{X}}_{(i)}$, $i = 1, ..., n$. The exact solution to this problem is well-know in both the Statistics [19] and computer vision fields [18] and is given by $\widehat{\mathbf{\Gamma}} = \boldsymbol{U}\boldsymbol{V}'$ where $\boldsymbol{U}$ and $\boldsymbol{V}$ are obtained from the singular value decomposition $\boldsymbol{Z}_2'\boldsymbol{Z}_1 = \boldsymbol{V}\mathbf{\Lambda}\boldsymbol{U}$. An important implementation detail of singular value decomposition for shape analysis is that to assure we have $\det(\widehat{\mathbf{\Gamma}}) = +1$ and hence a rotation matrix (as opposed to -1 and a reflection matrix), we can make instead $\widehat{\mathbf{\Gamma}} = \boldsymbol{U}\boldsymbol{R}\boldsymbol{V}'$ where $\boldsymbol{R}$ is the identity matrix except for the last diagonal element for which we use $\det(\boldsymbol{U}\boldsymbol{V}')$.

The GPA algorithm as described assumes the statistical model

$$\boldsymbol{X}_i = \beta_i(\boldsymbol{\mu} + \boldsymbol{E}_i)\mathbf{\Gamma}_i + \mathbf{1}_k\boldsymbol{\gamma}_i', \quad i = 1, ..., n \tag{10}$$

where $\boldsymbol{\mu}$ is the mean shape of the objects and the $k \times m$ matrix of errors $\boldsymbol{E}_i$ is such that $\text{vec}(\boldsymbol{E}_i) \sim (\mathbf{0}, \sigma^2 \boldsymbol{I}_{km \times km})$ where $\mathbf{0}$ is a vector of km zeroes and $\boldsymbol{I}_{km \times km}$ is the $km \times km$ identity. The model then assumes *isotropic variance*, i.e., the variance is the same at each landmark and in each of the m coordinates. Modification of GPA for the case of a general covariance matrix of the errors $\mathbf{\Sigma}$ requires a straightforward modification of the definition of the d_F distances minimized in (7) that accounts for $\mathbf{\Sigma}$. However, given that in general $\mathbf{\Sigma}$ is unknown and needs to be estimated there is no known registration algorithm which guarantees convergence in the non-isotropic case. Common practice is to initially set $\mathbf{\Sigma} = \boldsymbol{I}$, run GPA, then estimate $\mathbf{\Sigma}$ with

$$\widehat{\mathbf{\Sigma}} = \frac{1}{n}\sum_{i=1}^n \text{vec}(\boldsymbol{X}_i^p - \widehat{\boldsymbol{\mu}})\text{vec}(\boldsymbol{X}_i^p - \widehat{\boldsymbol{\mu}})'$$

run again GPA with the squared full procrustes distances in (7) replaced by the Mahalanobis squared distance $\text{vec}(\boldsymbol{X}_i)'\widehat{\mathbf{\Sigma}}^{-1}\text{vec}(\boldsymbol{X}_j)$ and iterate this process (but convergence is not guaranteed).

Model (10) implies that each object results from the rotation, scaling, and translation of the mean shape in the presence of random noise, i.e., similarity transformations of the mean shape observed with noise generate the observed profiles of the objects.

Example. Generalized Procrustes Registration. Suppose 10 cylindrical parts are manufactured. The parts have the same geometric specifications and were produced under homogeneous conditions. It is of interest to study the variability of the shapes (more on this below). The part design has a "notch", typical in parts that are used for assemblies. The 10 measured 2-dimensional shapes correspond to orthogonal contours obtained using a CMM at a fixed distance from the cylinder's origin. Each shape has $k = 200$ landmark measurements. We assume the matching landmark problem does not exist, so the labels between shapes correspond to each other. The original orientation of the parts, however, differs, and registration is necessary. Figure 5 shows ten such simulated contours before and after registration using the GPA algorithm (evidently, the noise has been exaggerated with respect to what actual measurements of real parts would look like). ■.

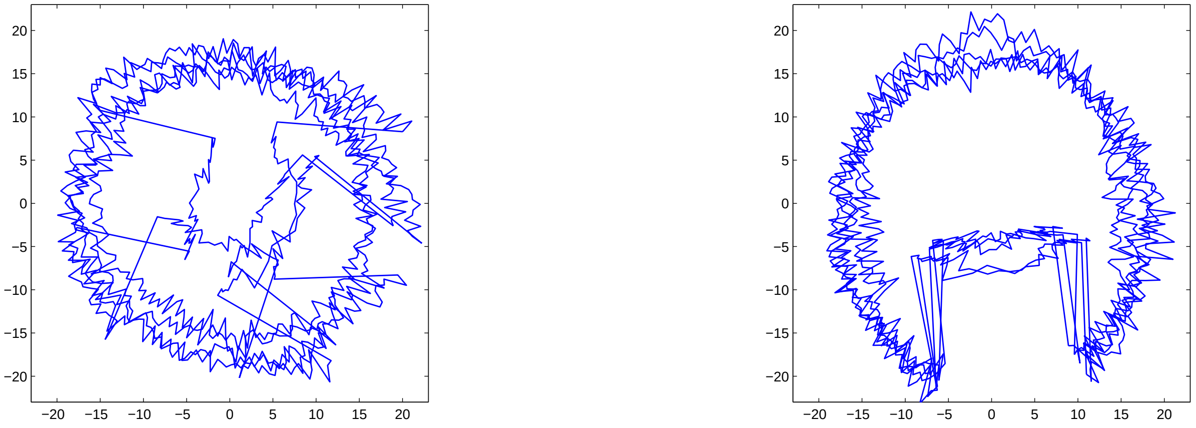

Figure 5: Example of GPA registration applied to the contours of 10 simulated "circular notched" parts, each with $k = 200$ (labeled) landmarks. Left: original, unregistered shapes. Right: registered shapes using GPA algorithm.

3.3 Tangent Space Coordinates

Once n configurations or profiles have been registered using GPA, the mainstream of the SSA literature (see e.g., [13, 15, 1]) recommends that further statistical analysis of shape variability and any desired inferences be made based on the resulting registered shapes $\boldsymbol{X}_i^p$ using the full Procrustes distances from the mean shape (or *pole*), called the *tangent space coordinates*. This is suggested in contraposition to working with the procrustes distances which are not linear. A Principal Component Analysis (PCA) is then recommended on the tangent space coordinates to better understand the directions in which the shapes are varying the most.

For a preshape $\boldsymbol{X}_i^p$ and mean shape $\widehat{\boldsymbol{\mu}}$, the tangent coordinates $\boldsymbol{v}_i$ are the distances along the tangent at the mean shape corresponding to the projection of $\boldsymbol{X}_i^p$ on $\widehat{\boldsymbol{\mu}}$ and are given by

$$\boldsymbol{v}_i = \left[\boldsymbol{I}_{(k-1)m} - \text{vec}\left(\frac{\widehat{\boldsymbol{\mu}}}{||\widehat{\boldsymbol{\mu}}||} \right) \text{vec}\left(\frac{\widehat{\boldsymbol{\mu}}}{||\widehat{\boldsymbol{\mu}}||} \right)' \right] \text{vec}\left(\frac{\boldsymbol{X}_i^p}{||\widehat{\boldsymbol{X}_i^p}||} \right), \quad i = 1, ...n. \tag{11}$$

The tangent coordinates are $(k-1)m$ vectors. If the centered configurations are used, then v_i is a km dimensional vector. An alternative approach which is close to the tangent coordinates if the preshapes are not too different from the mean shape (see Figure 6) is to use the *Procrustes residuals* $\boldsymbol{r}_i$ defined by

$$\boldsymbol{r}_i = \text{vec}(\boldsymbol{X}_i^p - \widehat{\boldsymbol{\mu}}), \quad i = 1, ..., n$$

that is, we work with the secants instead of the tangents. Again, for small differences about the mean, the conclusions of the analysis would be very similar. Regardless how one computes the tangent coordinates, either using (11) or using the approximate tangent coordinates $v_i \approx r_i$, the mainstream approaches to SSA recommend using a Principal Components Analysis (PCA) on the $\boldsymbol{v}_i$'s [15, 1]. The theoretical justification for this recommendation comes from work by Kent

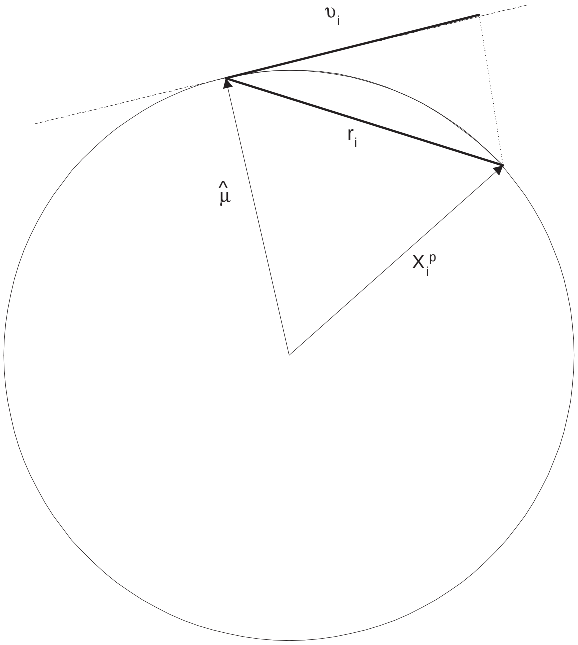

Figure 6: Tangent coordinates $\boldsymbol{v}_i$ and approximate tangent coordinates (secants) $\boldsymbol{r}_i = \text{vec}(\boldsymbol{X}_i^p - \widehat{\boldsymbol{\mu}})$.

and Mardia [25] who have shown that an isotropic distribution of the landmarks results in a isotropic distribution in the tangent space (given that small changes in a configuration matrix $\boldsymbol{X}$ induce an approximately linear change in the tangent coordinates v), and hence, PCA in tangent space is valid for shape analysis.

Let $\boldsymbol{V} = [\boldsymbol{v}_1, \boldsymbol{v}_2, ..., \boldsymbol{v}_n]$ be all the tangent coordinates of all the n objects under study. An estimate of the covariance matrix $\mathrm{Cov}(\boldsymbol{V}')$, giving the between-shape variances and covariances at the landmarks is given by

$$\boldsymbol{S}_v = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{v}_i - \overline{\boldsymbol{v}})(\boldsymbol{v}_i - \overline{\boldsymbol{v}})'$$

where $\overline{\boldsymbol{v}}$ is the average of the $\boldsymbol{v}_i$'s. This is a $(k-1)m \times (k-1)m$ matrix if the preshapes are only scaled and a $km \times km$ matrix if the centered preshapes are used instead. In the first case, the rank of this matrix is $p = M = (k-1)m - 1 - m(m-1)/2$ and in the latter case the rank is $p = M + 1$, since the mean is not lost. Let $\{\lambda_j\}_{j=1}^{p}$ and $\{\boldsymbol{e}_j\}_{j=1}^{p}$ be the p eigenvalues and eigenvectors of $\boldsymbol{S}_v$. Dryden and Mardia [13] suggest to compute

$$\boldsymbol{v}(c, j) = \overline{\boldsymbol{v}} + c\sqrt{\lambda_j}\boldsymbol{e}_j, \;\; j = 1, ..., p$$

for several values of c, say for $-6 < c < 6$.

One of the greatest advantages of shape analysis methods is visualization, as it takes place in a space that preserves the geometry of the objects. To visualize the principal components of the tangent coordinates, Dryden and Mardia [13] suggest to plot

$$\mathrm{vec}(\boldsymbol{X}_I) = \begin{bmatrix} \boldsymbol{H}' & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{H}' \end{bmatrix} [\boldsymbol{v}(c, j) + \mathrm{vec}(\widehat{\boldsymbol{\mu}})] \tag{12}$$

for all principal components j and for all multiplies c. These are the coordinates on the original shapes where the (registered) objects exist, and indicate the directions in which the principal components indicate movement –variability– around the mean shape (if the v_i's are km dimensional vectors, there is no need to premultiply times the block matrix of Helmert matrices). Just as in regular PCA, the percentage of variation explained by the jth principal component is given by $100\lambda_j/\sqrt{\sum_{j=1}^{p}\lambda_j}$. *Once the tangent coordinates have been computed, multivariate analysis techniques can be applied in the usual way* until the point where visualization using (12) is necessary.

Example. PCA of the circular notch shape data. Consider the 10 shapes shown in Figure 5. These shapes were simulated by superimposing sinusoidal variability along the circle at a second harmonic (inducing a bilobed shape) and inducing variability in the depth of the notch. Additional random normal bivariate variability was added at each landmark, which masks the first two sources of variability in such a way that they are not obvious to the eye. These sources of variability where added to demonstrate the power of PCA in the tangent space. Figure 7 shows the first two Principal Components in this example, which together account for more than 65% of the variability. Note how the first component is precisely the simulated bilobed shape and the second component refers to the depth of the notch. Remaining PC's do not show any obvious pattern. ■.

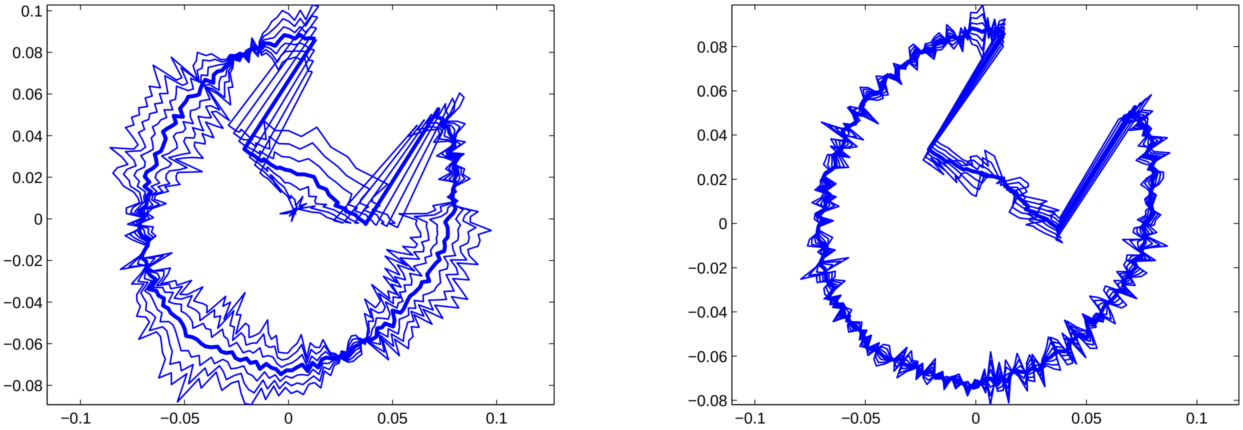

Figure 7: Example of Principal Components Analysis applied to the data of the 10 circular notched parts. Left: first principal component detects a sinusoidal variation, and accounts for 56.7% of the variability. Right: second principal component, which corresponds to variation in the depth of the notch, accounts for 8.3% of the variability. In these plots, values of $c \in (-3, 3)$ where used. Dark line is the mean shape.

Performing a PCA on the tangent coordinates is of value when one is interested in analyzing how the *variability* of the shapes behaves around the mean. For analyzing the effect of factors (varied during an experiment) on the *mean* shape (as required when conducting manufacturing experiments that may improve the shapes of parts produced by a process) one needs to perform an Analysis of Variance. This was first discussed by Goodall [15] (see also Dryden and Mardia [13]) for the one-way case and studied in the two-way layout case, with application in manufacturing, by Del Castillo and Colosimo [11].

4 Further work

The methods presented in this chapter assume all parts contain the *same* number of corresponding landmarks, or locations of interest. If there is a different number of landmarks between 2 objects, the ICP (iterative closest point) algorithm [4, 35] has been proposed to obtain the same number of corresponding landmarks between the parts. If the landmarks do not correspond, a matching algorithm such as the context-labeling algorithm of section 2 could be applied after the ICP algorithm. An alternative to the use of the ICP algorithm [3] is to simply add dummy landmarks to the smallest landmark matrix to get $\max(k_1, k_2)$ landmarks, and assign a large cost between these points and all others (this is also a mechanism to handle outlier landmarks, since they would be matched to the dummy points).

A matching algorithm notably different than the one presented in section 2 has been proposed recently by Green and Mardia [17]. It also applies to the case of 2 objects. Other possibility is to use the 2-dimensional context histograms, but use a statistic other than the χ^2 used here, to measure distances (costs) between two multivariate distributions, e.g., a 2-dimensional Kolmogorov-Smirnov test or other recent alternatives (e.g., that in [32]). Such approach would still use the weighted matching LP formulation presented here, but with a different way to get the cost matrix $\boldsymbol{C}$. Even in the Belongie et al. [3] approach, it is not clear how to best scale the $\boldsymbol{X}$ matrices, how many bins to use in each dimension, or what is the best way to measure angles for differently oriented objects in order to achieve effective rotation invariance. An interesting embellishment to the landmark matching algorithm [3] is to iterate the matching algorithm with an algorithm for the estimation of the registration transformation between the objects. This may result in better matching (and hence, registration) because the initial matching may be sensitive to the different orientations of the parts due to the ambiguities mentioned earlier about defining the histogram resolution. These authors suggested to use Thin Plate Spline transformations, popular also in the area of Morphometrics, as opposed to the GPA algorithm considered here. A similar iterative procedure could be attempted with the context labeling algorithm and the GPA algorithm applied iteratively. A recent description of the matching problem from a Computer Vision perspective is the book by Davies et al. [10].

As mentioned earlier, a generalization of 2-object matching methods to the case of n objects is desirable, since once labeled (corresponding) landmarks are available (assuming same number of landmarks in each object), the SSA methods presented herein can be implemented. The advantages and disadvantages of the "context labeling" method compared to the Green/Mardia [17] approach need to be investigated.

In this paper we did not discuss tests for comparing the mean shapes between two or more populations, which can be done using ANOVA methods applied to the shapes. For more information on this topic, see [11].

Most of the work on SSA has focused on 2D shapes. Extensions to the 3-dimensional case are evidently practical (but the landmark matching problem becomes more difficult). The context labeling approach presented here has been extended recently by Frome et al. [12] to the 3-D case using 3-D histograms. The implementation details mentioned above remain and need to be studied). For 3-D objects, GPA and the PCA can be done without any change, but visualization of the PCA's is challenging if k (no. of landmarks) is large.

Finally, some authors (e.g., Lele and co-workers [28]) have proposed using the inter-landmark Euclidean distance matrix $[d_{ij}]$ to make inferences on the shapes of objects, with application to testing for the difference between shapes. Lele suggests using the GPA algorithm to estimate the mean shape, but debate exists about how to estimate the covariance matrix of the landmarks in the non-asitropic case. This series of methods do not have an easy way to visualize the results, and require more information ($\begin{pmatrix} k \\ 2 \end{pmatrix}$ distances instead of km), although this information is implicit in the $k \times m$ matrix $\boldsymbol{X}$. In addition, there seems to be no counterpart to the PCA analysis of variability in distance-based methods. There is considerable debate about which method is more powerful to detect differences in shapes, and it is of interest to compare distance-based methods with those studied in [11] for a variety of shapes of relevance in manufacturing, since the power of these methods appears to depend on the shape in question. Dryden and Mardia [13] present a good overview of distance-based methods.

Acknowledgement. This research was partially funded by the National Science Foundation grant CMI-0825786 and a grant from the Ministry of Science of Italy.

Appendix. Computer implementation of the Landmark Matching and GPA/PCA algorithms.

Matlab programs that perform the computations required for the context labeling algorithm of section 2 and for the GPA algorithm, including visualization of PCA's, were written for this research and can be downloaded from

http://www2.ie.psu.edu/Castillo/research/EngineeringStatistics/software.htm.

The programs posted contain several programs for statistical shape analysis. Two of the programs are related to what is discussed in the present paper: `ContextLabeling.m`, which implements the context labeling algorithm presented in section 2 for two 2-dimensional objects, and `GPA23.m`, which implements the generalized Procrustes algorithm (assuming isotropic variance), and performs, if desired, the Principal Component Analysis in tangent space, including the corresponding visualization.

References

[1] Adams, D.C., Rohlf, F.J., and Slice, D.E., "Geometric morphometrics: ten years of progress following the "revolution"", *Italian Journal of Zoology*, 71, pp. 5-16, (2004).

[2] American Society of Mechanical Engineers, *ASME Y14.5M-1994, Dimensioning and Tolerancing*, ASME, (1994).

[3] Belongie, S., Malik, J., and Puzicha, J., "Shape Matching and Object Recognition Using Shape Contexts", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(24), pp. 509-522, (2002).

[4] Besl, P.J., and McKay, N.D., "A method for registration of 3-D shapes", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2), (1992).

[5] Bookstein, F.L., "Size and shape spaces for landmark data in two dimensions", *Statistical Science*, 1, pp. 181-242, (1986).

[6] Bookstein, F.L., "Landmark methods for forms without landmarks: morphometrics of group differences in outlien shape", *Medical Image Analysis*, 1(3), pp. 225-243, (1996/1997).

[7] Chui, H., and Rangarajan, A., "A new algorithm for non-rigid point matching", *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 44-51, (2000).

[8] Colosimo, B.M., Pacella, M., Semeraro, Q., "Statistical Process Control for Geometric Specifications: On the Monitoring of Roundness Profiles", *Journal of Quality Technology*, 40(1), pp. 1-18, (2008).

[9] Colosimo, B.M., and Pacella, M., "On the use of principal component analysis to identify systematic patterns in roudnness profiles", *Quality and Reliability Engineering International*, 23, pp. 707-725, (2007).

[10] Davies, R., Twining, C., and Taylor, C., (2008), *Statistical Models of Shape, Optimisation and Evaluation*, Springer-Verlag, London.

[11] Del Castillo, E., and Colosimo, B.M., "Statistical Shape Analysis of Experiments for Manufacturing Processes", accepted for publication in *Technometrics*, Engineering Statistics Laboratory Technical paper, Penn State University, (2009).

[12] Frome, A.,Huber, D., Kolluri, R., Bulow, T., and Malik, J. "Recognizing objects in range data using regional point descriptors". *Proc. 8th Europ. Conf. Comput. Vision*, 3, pp. 224237, (2004).

[13] Dryden, I.L., and Mardia, K.V., *Statistical Shape Analysis*, Chichester, John Wiley & Sons, (1998).

[14] Gold, S., Rangarajan, A., Lu, C.-P., Pappu, S., and Mjolsness, E., "New Algorithms for 2D and 3D Point Matching: Pose Estimation and Correspondence", *Pattern Recognition*, 31(8) (1998).

[15] Goodall, C., "Procrustes Methods in the Statistical Analysis of Shape", *J. of the Royal Statistical Society, B*, 53, 2, pp. 285-339, (1991).

[16] Gower, J.C., "Generalized Procrustes Analysis", *Psychometrika*, 40,1, pp. 33-51, (1975).

[17] Green, P.J., and Mardia, K.V., "Bayesian alignment using hierarchical models, with applications in protein bioinformatics", *Biometrika*, 93(2), pp. 235-254, (2006).

[18] Horn, B.K.P., Hilden, H.M., and Negahdaripour, S., "Closed-form solution of absolute orientation using orthonormal matrices", *Journal of the Optical Society of America A*, 5, pp. 1127-1135, (1988).

[19] Jackson, J.E., *A user's guide to principal components*, New York: John Wiley & Sons, (2003).

[20] Kang, L., and Albin, S.L., "On-line monitoring when the process yields a linear profile", *Journal of Quality Technology*, 32, pp. 418-426, (2000).

[21] Kendall, D.G., "Shape Manifolds, procrustean metrics, and comples projective spaces". *Bull. London math. Soc.*, 16, pp. 81-121, (1984).

[22] Kendall, D.G., "A Survey of the Statistical Theory of Shape", *Statistical Science*, 4,2, pp. 87-89, (1989).

[23] Klingenberg, C.P., McIntyre, G.S., "Geometric Morphometrics of Developmental Instability: Analyzing Patterns of Fluctuating Asymmetry with Procrustes Methods", *Evolution*, 52(5), pp. 1363-1375, (1998).

[24] Klingenberg, C.P., and Monteiro, L.R., "Distances and directions in multidimensional shape spaces: implications for morphometric applications", *Systematic Biology*, 54(4), pp. 678-688, (2005).

[25] Kent, J.T., and Mardia, K.V., "Shape, Procrustes tangent projections and bilateral symmetry", *Biometrika*, 88,2, pp. 469-485, (2001).

[26] Krulikowski, A., *Geometric Dimensioning & Tolerancing*, (A companion to the ASME Y14.5M-1994 dimensioning & tolerancing standard), Effective Training Inc., Westland, MI, (1996).

[27] Langron, S.P., and Collins, A.J., "Perturbation Theory for Generalized Procrustes Analysis", *Journal of the Royal Statistical Society*, B, 47(2), pp. 277-284, (1985).

[28] Lele, S.R., and Richtsmeier, *An invariant approach to statistical analysis fo shapes*, Boca Raton, FL: Chapman & Hall/CRC, (2001).

[29] Mardia, K.V., discussion to "A survey of the statistical theory of shape" by D.G. Kendall. *Statistical Science*, 4, pp. 108-111, (1989).

[30] Papadimitrou, C.H., and Steiglitz, K., *Combinatorial Optimization, Algorithms and Compelxity*, Prentice Hall, Englewood Cliffs, NJ, (1982).

[31] Rohlf, F.J., and Slide, D., "Extension of the procrustes method for the optimal superimposition of landmarks", *Syst. Zoology*, 39(1), pp. 40-59, (1990).

[32] Rosenbaum, P.R., "An exact distribution-free test comparing two multivariate distributions based on adjacency", *J. R. Statist. Soc. B*, 67(4), pp. 515-530, (2005).

[33] Ten Berge, J.M.F., "Orthogonal Procrustes Rotation for Two or More Matrices", *Psychometrika*, 42-2, pp. 267-276, (1977).

[34] Woodall, W.H., Spitzner, D.J., Montgomery, D.C., and Gupta, S., "Using control charts to monitor process and product quality profiles", *Journal of Quality Technology*, 36(3), pp. 309-320, (2004).

[35] Zhang, Z., "Iterative point matching for registration of free-form curves", Reports de Recherche no. 1658, IRIA, Sophia Antipolis, France.