A Trainable Method of Parametric Shape Description

T.F.Cootes, D.H.Cooper, C.J.Taylor and J.Graham
Department of Medical Biophysics
University of Manchester
Oxford Road
Manchester M13 9PT

Abstract

We have developed a trainable method of shape representation which can automatically capture the invariant properties of a class of shapes and provide a compact parametric description of variability. We have applied the method to a family of flexible ribbons (worms) and to heart shapes in echocardiograms. We show that in both cases a natural parameterisation of shape results.

1 Introduction

Shape models have been used widely to achieve robust interpretation of complex images. They allow image evidence to be organised into plausible interpretations which can then be verified [1–3]. We are interested in the class of problems where shapes are variable. Important examples are the inspection of complex manufactured assemblies, where relative motion between subparts is possible, and medical image interpretation where biological variation is present. In such applications it is generally the case that some aspects of shape are invariant whilst others are subject to constrained variability. The problem of adequately modelling such behaviour, in a general way, has not been solved. We have previously described a method of shape representation based on modelling the statistical distributions of chord lengths between control points placed in a consistent manner on each shape in a training set [4, 5]. The objectives of the work we describe here were to significantly develop this basic idea to automatically:

1. Make shape invariants more explicit
2. Identify and parameterise the significant degrees of freedom in a set of training shapes.

Our principle motivation was the wish to develop efficient methods of image interpretation based on flexible template matching. This requires generalisation from a training set, to generate plausible instances of shapes which satisfy the constraints exhibited in the training set, yet can be controlled using a small number

of parameters. The basic approach is as follows;

1. Gather chord statistics from the training set.

2. Calculate invariant and covariant sets of chords.

3. Generate new sets of chords from mean chords + weighted sums of covariant sets. (Varying the weights varies the form of the shape reconstructed.)

4. Reconstruct shape from new set of chords.

Shapes are generated by varying the lengths of chords around their mean values in such a way that the shape invariants are maintained. This is achieved by deriving, from the training set, a form of relationship between the shape controlling parameters and the chord lengths which is guaranteed (to a first approximation) not to modify the invariant properties of the chord set. A shape is reconstructed by finding the positions of control points which are most consistent with the given set of chord lengths.

2 Method

Taylor and Cooper [4] describe a method of shape representation called the Chord Length Distribution (CLD). The CLD is trained on a set of s shapes which are represented by a set of n–vertex polygons. For each polygon in the training set the m = ½n(n–1) chord lengths R_i between all pairs of points are calculated, giving a m–vector $\underline{R} = \{ R_i \}$.

A set of chord lengths is said to be 'Euclidean' in two dimensions if it can be generated from a real set of points, the vertices of an n–gon [6]. The lengths of many chords are inter–related, so one cannot adjust them arbitrarily and retain a Euclidean set. It is possible to estimate the correlation between pairs of chords by calculating their covariance over the training set.

The covariance (C_{ij}) between pairs (i.j) of chords is given by :

$$C_{ij} = \frac{1}{s} \sum_{t=1}^{s} (R_i^{(t)} - \mu_i)(R_j^{(t)} - \mu_j) \qquad (1)$$

where s is the number of shapes in the training set,
$R_i^{(t)}$ is the i'th chord length of the t'th shape in the training set,
μ_i is the mean length of R_i.

This gives a m x m covariance matrix $\underline{C} = \{C_{kl}\}$ for a given training set. We can find the (m) normalized eigenvectors, r_k, of $\underline{C}$ such that

$$\underline{C}\underline{r}_k \; = \; \lambda_k \underline{r}_k \qquad\qquad (\; \underline{r}_k^T\underline{r}_k \; = \; 1 \;) \qquad\qquad\qquad (2)$$

$$(\; \lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \; ... \; \geq \lambda_m \;)$$

These eigenvectors are combinations of variations in chord lengths which are linearly independent. If we make the assumption that the dependencies between chord lengths are linear, the eigenvectors may be treated as a totally independent parameterisation of shape variability. The eigenvectors corresponding to large eigenvalues represent significant degrees of freedom in the family of shapes from which statistics have been obtained, whilst the vectors with small (or zero) eigenvalues represent shape invariants. Although the assumption of linear dependence does not strictly hold in the system we describe, it is a reasonable approximation, particularly where shape variability is modest.

The eigenvectors are mutually orthogonal and span the m–dimensional chord space. If $\underline{r}$ is the (m x m) matrix of eigenvectors, $\underline{r} = (\; \underline{r}_1 \mid \underline{r}_2 \mid \underline{r}_3 \mid \; ... \; \mid \underline{r}_m \;)$, any set of chords, $\underline{R}$, can be written;

$$R = \underline{\mu} + \underline{r}\underline{b} \qquad\qquad\qquad (3)$$

where $\underline{b}$ is a (m x 1) column vector, $(\; b_1 \; b_2 \; b_3 \; ... \; b_m \;)^T$,

$$\underline{b} = \underline{r}^T (\underline{R} - \underline{\mu}) \qquad\qquad\qquad (4)$$

(Since the columns of $\underline{r}$ are orthogonal, $\underline{r}^T = \underline{r}^{-1}$)

Each shape in the training set can thus be represented by a vector, $\underline{b}$, in a new m–dimensional space. It can be shown that in this space the variance of the parameter b_k over the set of training shape $\underline{b}$–vectors is λ_k, the kth eigenvalue of the covariance matrix, $\underline{C}$ [7]. Thus the vector of chord deviations, $\underline{r}_k$, explains λ_k of the variance of the chords in the training set.

We wish to choose a reduced set of chord vectors which can explain most of the variation in shape. This will allow us to generate shapes similar to those in the training set by varying only a small number of parameters. The variance explained by the first t eigenvectors is

$$V_t \; = \; \sum_{k=1}^{t} \lambda_k \qquad\qquad\qquad (5)$$

If t is chosen such that V_t is a suitably large proportion of V_m, the total variance, the first t eigenvectors will then be able to explain most of the variability in the training set. Almost Euclidean sets of chords $(\underline{R})$ can be generated by taking the mean chord lengths and adding weighted combinations of the first t eigenvectors corresponding to the large eigenvalues.

$$\underline{R} = \underline{\mu} + \sum_{k=1}^{t} b_k \underline{r}_k$$

$$\underline{R} = \underline{\mu} + \underline{r'}\underline{b'} \qquad (6)$$

Where $\underline{r}'$ is the (m x t) matrix of eigenvectors; $\underline{r}' = (\underline{r}_1 \mid \underline{r}_2 \mid ... \mid \underline{r}_t)$
$\underline{b}'$ is a (t x 1) (column) vector of parameters, b_k.

If a shape is reconstructed from the new set of chords, $\underline{R}$, the parameters b_k (k = 1 .. t) will control the variations in the shape.

3 Generating Polygons From Chord Sets

Because the above method uses a linear approximation to the possibly non–linear relationship between chord lengths, and is statistical in nature, the sets of chords produced as $\underline{b}'$ is varied may not be Euclidean in 2–D (will not precisely correspond to a polygon). We define the polygon which best fits a set of chords ($\underline{R}_0$) constructed from a set of parameters ($\underline{b}_0$) as the set of points $\underline{x} = \{(x_i, y_i)\}$ (i = 0..n–1) which minimise the weighted sum ;

$$F(\underline{x}) = \sum_{i=1}^{m} w_i (R_i - R_{i,0})^2 \qquad (7)$$

where w_i is a weight,

$w_i = 1/\sigma_i$ if $\sigma_i > 0.01$,

$w_i = 100$ otherwise.

(σ_i is the standard deviation of the i'th chord over the training set.)

This weighting scheme ensures that those chords which are invariant in the training set have the same length in the reconstructed shape. Such chords have small standard deviations, σ_i, so will have large weights, w_i, encouraging $R_i = R_{i0}$.

The minimisation is currently achieved using a 'steepest descent' optimisation method [8].

4 Experimental Results

We have tested the method described above by applying it to two shape parameterisation problems. The first involves a synthesised set of "worm" shapes whose shape invariants and modes of variability are both known. The second relates to a practical application in medical image interpretation and involves the outline of the left ventricle as seen in echocardiograms. In both cases a training set of shapes has been used to automatically generate a parametric model with only a few degrees of freedom.

4.1 "Worms"

A set of 21 "worm" shapes, each described by 12 boundary points (Figures 1 and 2), was used to train the system. The worms varied in length and axial curvature but were of constant width. This example represents a challenging problem for the method; the important invariant is an axial symmetry which we wish the system to discover from the example shapes. At the same time we wish the parameterisation to allow significant variations in length and curvature.
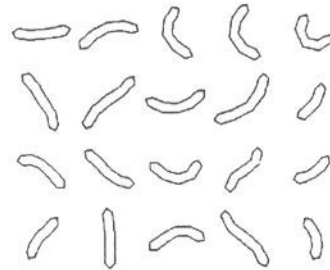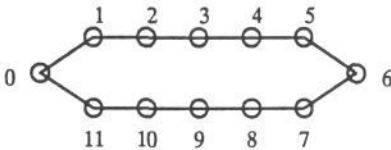

Figure 1. Labelling of 12 Point "worm" polygon used in training.

Figure 2. Examples of 12 point worms in training set.

The eigenvalues of the 66 x 66 covariance matrix, $\underline{C}$, are show in Table 1. These results suggest that the worms have two significant degrees of freedom. The ranges of parameters b_1 and b_2 for the shapes in the training set (derived using Equation 4) are shown in Figure 3. The apparently random scatter suggests b_1 and b_2 can be varied independently to generate new shapes.

Eigenvalue	% total sum of eigenvalues
$\lambda_1 = 959$,	75%
$\lambda_2 = 216$,	17%
$\lambda_3 = 44$.	3%
$\lambda_4 = 30$	2%
$\lambda_5 = 25$	2%
$\lambda_i < 5 \quad i > 5$.	1%

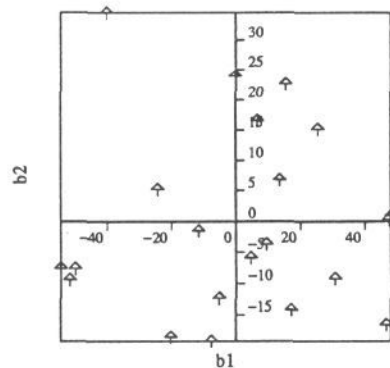Table 1. Eigenvalues of Covariance Matrix Derived from "Worm" Data Set.

Figure 3. Scattergram of b_1 vs b_2 for Shapes in the Worm Training Set.

Examples of shapes generated by varying b_1 and b_2 are shown in Figure 4. The most significant eigenvector, $\underline{r}_1$, appears to affect the length of the worm. The next eigenvector, $\underline{r}_2$, affects the degree of overall curvature in the worm. In both cases

axial symmetry is preserved. The third parameter determines the amount to which the ends of the worms curve in opposite directions, but its effects are small as there were few examples of this in the training set.
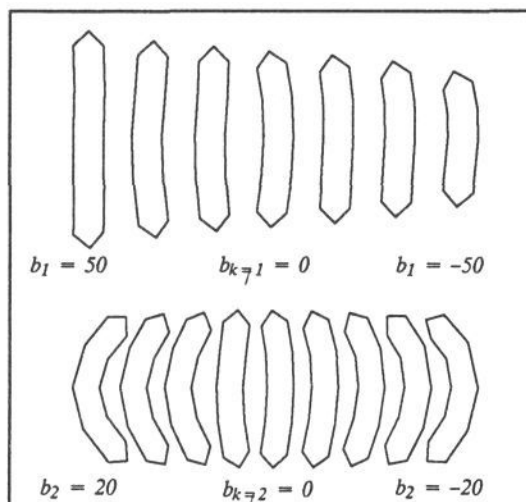

$b_1 = 50$ $b_{k \neq 1} = 0$ $b_1 = -50$

$b_2 = 20$ $b_{k \neq 2} = 0$ $b_2 = -20$

Figure 4. The effects of varying the parameters corresponding to the largest two eigenvalues.

4.2 Heart Data Set

A project currently being undertaken by Hill and Taylor [9] involves finding the boundary of the left ventricle in echocardiograms, ultrasound images of the heart. A 'hand crafted' parameterised model of the boundary is matched to the image using a Genetic Algorithm. We have investigated the possibility of constructing the model automatically from a set of examples. A training set was generated by manually drawing the heart boundary on each of 66 images (Figure 6). Each boundary was represented by an 18–vertex polygon . Four control points were placed on each boundary by hand, the 14 other points were equally spaced along the boundary between the control points.

The eigenvalues of the covariance matrix derived from the shapes in the training set are shown in Table 2. The ranges of parameters b_1 and b_2 for the shapes in the training set (derived using Equation 4) are shown in Figure 5.

Eigenvalue	% total sum of eigenvalues
$\lambda_1 = 24076$	91%
$\lambda_2 = 952$	4%
$\lambda_3 = 465$	2%
$\lambda_4 = 330$	1%
$\lambda_5 = 209$	1%

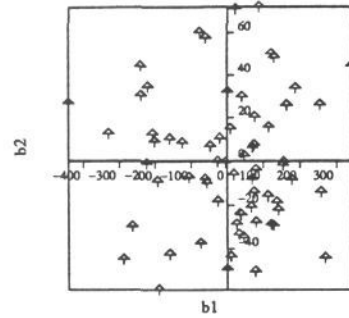Table 2. Eigenvalues of Covariance Matrix Derived from 'Heart' Data Set.

Figure 5. Scattergram of b1 vs b2 for Heart Shapes in Training Set.

The results of varying the first four parameters are shown in Figure 7. The parameter associated with the largest eigenvalue, b_1, controls the scale of the shape. The second parameter, b_2, seems to affect the width of the shape. The third and fourth parameters seem to affect the width and the form of the base of the shape, which corresponds to the opening and closing of the mitral valve in the heart.
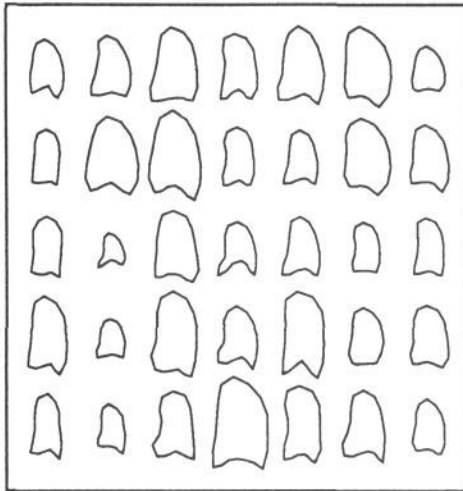

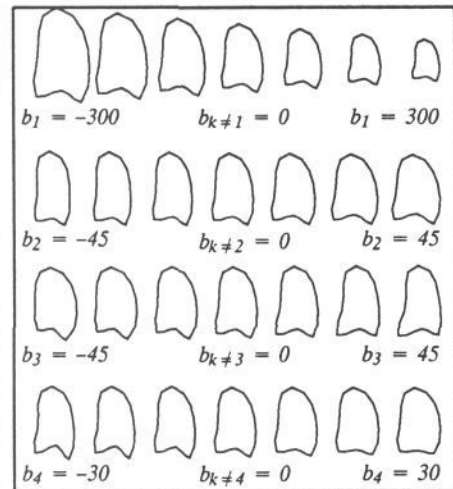Figure 6. Examples from the Training Set of 18–Vertex Heart Shapes.

Figure 7. Effect of varying the parameters associated with the four largest eigenvalues.

5 Discussion

The method has been applied to two cases and in both gives a parameterised model of shape. The resulting models have both been used to find boundaries in noisy images.

For some types of variability (eg rotation of one subpart around another) the assumption of linear dependence between chords does not hold. This leads to correlated b–vectors for members of the training set, so it is not reasonable to choose the elements of the b–vectors independently when seeking to generate a new instance of the model. We are looking at ways of dealing with this.

Although the examples given above are of shapes, it is the position of the points which are modeled. These can represent vertices of polygons or the positions of subparts equally easily, allowing spatial relationships between parts to be modelled.

The method could be used to model the image variability due to small changes in viewpoint and lighting conditions in industrial inspection.

The choice of points in the training set is important. Each point must be in a position which can be reproduced in each training shape. Choosing points can be a way of using human expertise in the training phase, though it would be useful to automate the procedure of positioning the points on training images as much as possible.

6 Acknowledgements

7 References

[1] Chin R.T., Dyer C.R. Model–Based Recognition in Robot Vision. Computing Surveys, Vol 18, No 1, 1986

[2] Grimson, W.E.L.,Object Recognition by Computer : The Role of Geometric Constraints. The MIT Press, Cambridge Massachusetts, 1990

[3] Cooper D.H., Bryson N., Taylor C.J. An Object Location Strategy using Shape and Grey–level Models. Image and Vision Computing , Vol 7, No 1, pp 50–56, 1989

[4] Taylor C.J. & Cooper D.H., Shape Verification using Belief Updating, Proceedings BMVC 1990

[5] Cooper D.H., Taylor C.J., Graham J. & Cootes T.F. Locating Overlapping Flexible Shapes Using Geometrical Constraints. (This Volume).

[6] Gower J.C. Euclidean Distance Geometry. Math. Sci., Vol.7, pp1–14 1982

[7] Fukunaga K. and Koontz W.L.G. Application of the Karhunen–Loeve Expansion to Feature Selection and Ordering. IEEE Trans. on Computers, Vol. C19, No.4, April 1970.

[8] Gill P., Murray W. and Wright M. Practical Optimisation. Academic Press Inc., 1981

[9] Hill A. and Taylor C.J. Model Based Image Interpretation Using Genetic Algorithms. (This Volume)