Improving Specificity in PDMs using a Hierarchical Approach

Tony Heap and David Hogg

School of Computer Studies, University of Leeds, Leeds, UK, LS2 9JT

`ajh@scs.leeds.ac.uk`

Abstract

The Point Distribution Model (PDM) has proved useful for many tasks involving the location and tracking of deformable objects. A principal limitation is non-specificity; in constructing a model to include all valid object shapes, the inclusion of some invalid shapes is unavoidable due to the linear nature of the approach.

Bregler and Omohundro [2] describe a 'piecewise linear' method for applying constraints within model shape space, whereby principal component analysis is used on training data clusters in shape space to generate lower dimensional overlapping subspaces. Object shapes are constrained to lie within the union of these subspaces, thus improving the specificity of the model.

This is an important development in itself, but its most useful quality is that it lends itself to automated training. Manual annotation of training examples has previously been necessary to ensure good specificity in PDMs, requiring expertise and time, and thus limiting the amount of training data that can feasibly be collected. The use of shape space constraints means that such accurate annotation is unnecessary, and automated training becomes significantly more successful.

In this paper we expand on Bregler and Omohundro's work, suggesting an alternative representation for the linear pieces, and showing how a two-level hierarchy in shape space can be used to improve efficiency and reduce noise. We perform an evaluation on both synthetic and (automatically trained) real models.

1   Introduction

Models of shape are used widely in computer vision; image features can be located, tracked or classified using *a priori* knowledge of object shape. Many objects are non-rigid, requiring a *deformable* model in order to capture shape variability.

One such model is the Point Distribution Model (PDM) [3]. An object is modelled in terms of landmark points positioned on object features, and at regular intervals in between. By identifying such points on a set of training examples, a statistical approach (principal component analysis, or PCA) can be used to discover the mean object shape, and the major modes of shape variation.

The standard PDM is based purely on linear statistics (the PCA assumes a Gaussian distribution of the training examples in shape space). For any particular mode of variation, the positions of landmark points can vary only along straight lines; non-linear variation is achieved by a combination of two or more modes. This situation is not ideal, firstly because the most compact representation of shape variability is not achieved, and secondly because implausible shapes can occur, when invalid combinations of deformations are used.

Attempts have been made to combat this problem. Sozou *et al*'s Polynomial Regression PDM [9] allows landmark points to move along combinations of polynomial paths. Heap and Hogg's Cartesian-Polar Hybrid PDM [6] makes use of polar coordinates to model bending deformations more accurately. Sozou *et al* [10] have also investigated using a multi-layer perceptron to provide a non-linear mapping from shape parameters to shape.

All these approaches give some improvement over the linear PDM, but they have their limitations. The first two can only model certain types of non-linear deformation (polynomial and rotational respectively). The perceptron method can model more general non-linear deformations in one dimension, but performance is poor in cases where there is more than one degree of deformational freedom.

The common feature of previous approaches is some form of 'linearising' mapping of the shape space onto another uniform space. In some cases such a mapping cannot exist; for example when the distribution of valid shapes forms a region which is hollow, has changing dimensionality, or is discontinuous.

Bregler and Omohundro [2] describe a method for approximating an arbitrary *surface* within an n-dimensional space, using samples taken from it. The training examples are divided into (overlapping) clusters, and a PCA is performed separately on each cluster. This produces a set of locally-linear 'patches', the union of which gives the required approximation to the surface.

This technique applies directly to object shape modelling. If the training samples are examples of valid object shapes then the surface produced is the region of valid shapes. Bregler models the shape of lips in this way.

The piecewise-linear approach has also been touched on by Ahmad *et al* [1]. They built a multi-gesture hand model consisting of 5 sub-PCAs (one for each gesture) using a weighted combination of the training examples, and experienced promising results in terms of tracking performance. Automation of the model building process was not considered; both the collection of training data and the determination of cluster membership were undertaken manually.

In this paper we extend the ideas of Bregler and Omohundro. We describe alternative treatments of the locally-linear patches and union operations, and highlight some of the design choices which must be made. We also show how it is possible to use a two-level hierarchical approach to improve efficiency and reduce noise.

We apply the technique to synthetic data containing non-linear deformation (an anglepoise lamp), and on automatically-collected real data (hand shapes).

2 The Point Distribution Model

A PDM is built purely from the statistical analysis of a number of examples of the object to be modelled [3]. Given a collection of training images of an object, the Cartesian coordinates of N strategically-chosen landmark points are recorded for each image. Training example e is represented by a vector $\mathbf{x_e} = (x_{e1}, y_{e1}, \ldots, x_{eN}, y_{eN})$ (for a 2D model).

The examples are aligned (translated, rotated and scaled) using a weighted least squares algorithm, and the mean shape $\overline{\mathbf{x}}$ is calculated by finding the mean position of each landmark point. The modes of variation are found using Principal Component Analysis (PCA) on the deviations of examples from the mean, and are represented by N orthonormal 'variation vectors' $\mathbf{v_j}$. An object shape $\mathbf{x}$ is generated by adding linear combinations of the t most significant variation vectors to the mean shape:

$$\mathbf{x} = \overline{\mathbf{x}} + \sum_{j=1}^{t} b_j \, \mathbf{v_j} \tag{1}$$

where b_j is the weight for the jth variation vector. Generally, most important deformation is captured by only a few variation vectors; the rest represent noise in the training data. By choosing $t \ll 2N$, we extract only the important deformations, discarding noise, and thus compactly capture object shape and variation.

2.1 Limitations of the Linear PDM

A good deformable model should be *accurate, specific* and *compact*. An accurate model includes all valid shapes. A specific model excludes all invalid shapes. A compact model uses the smallest number of parameters possible to describe a shape (ie. its dimensionality approaches the natural deformational dimensionality of the object being modelled).

Model shape can be described in terms position within an n-dimensional *shape space*. In the case of a PDM the dimensions are the x and y coordinates of every landmark point. Within the shape space there is generally a continuous region which corresponds to *valid* shapes; in this paper we refer to this as the valid shape region, or VSR.

The Linear PDM assumes that the set of all valid shapes forms a Gaussian distribution about some mean point in the shape space, representing the VSR as being bounded by a hyperellipsoid. In some cases, especially when model landmarks have been chosen strategically, this approximation is sufficient to produce a satisfactory model which is both compact and specific. However, in many real objects, non-linear deformations (such as bending or pivoting) are a natural occurrence. The Linear PDM is forced to model non-linear deformations by the combination of two or more linear deformations. Such models are not compact because the dimensionality is increased, and not specific because invalid shapes can be produced via an invalid combination of linear deformations (see Figure 4 for examples).

There are various techniques one can use to transform the shape space in such a way as to linearise the VSR [9, 10, 6]. In these approaches there is always a notion of a base shape (usually the mean shape) and a fixed number of independent modes of variation, valid over a fixed, continuous range. However, in some cases the VSR is *not* linearisable in a simple manner. A VSR can in theory have an arbitrary topology and its dimensionality can vary over the shape space. This is perhaps more common than one might think; for example, in building a model of three hand gestures and the transitions amongst them, the VSR takes the form of a hollow triangle. A method is required for representing *any* possible VSR.

3 A Hierarchical PDM

Bregler and Omohundro [2] describe a solution to this problem, whereby a constraint surface is constructed within shape space, using a union of lower dimensional subspaces. We have altered and extended this technique by considering a two-level hierarchical approach, and by substituting the hyperplane subspaces for bounded regions.

The key to the approach is that a complex, non-linear region approaches linearity locally under magnification, and hence can be approximated by a combination

of a number of smaller linear subregions. To build the subregions, a k-means cluster analysis is performed on the training data in shape space to find a number of prototypes. For each prototype, a number of nearest neighbours are taken from the training set and a PCA is performed on them. A subregion is produced which is centred on the cluster mean (*not* necessarily the same as the prototype), and bounded by a hyperellipsoid with a Malhalanobis radius of some specified value M (similar to a standard Linear PDM). The VSR is then represented as the union of these subregions (there are some subtleties; these are discussed later). Figure 1 demonstrates the process.
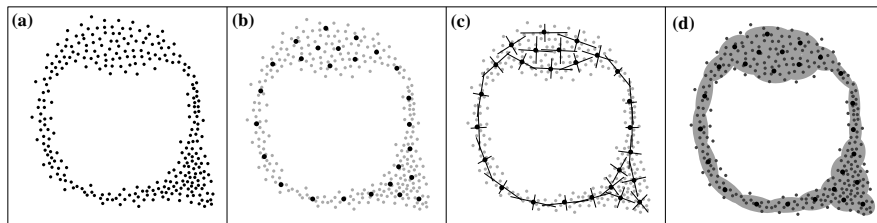

Figure 1: Building a valid shape region from linear patches; (a) training data projected into a 2D shape space, (b) k-means cluster centres, (c) principal axes and (d) the valid shape region.

In the case of the PDM, the shape space generally has upwards of 100 dimensions (twice the number of model landmark points). For this reason it is useful to adopt a two-level hierarchical approach; an initial *global* PCA is performed on the training data in order to produce a lower dimensional space. The linear subregions are then constructed in this new space instead of the high dimensional shape space. The reduced dimensionality decreases computation times substantially, and noise from outliers is reduced due to the removal of insignificant modes of variation by the global PCA.

For the VSR to be of practical use, it must be possible to apply what is known as the 'nearest point' query: "Given a general point in shape space, where is the nearest point in the VSR?" This then facilitates the application of constraints to any given shape.

The simplest (but not only) way to apply these constraints is to find the closest (Euclidian distance) cluster mean and constrain the point to be within the associated subregion. The point should ideally be moved to the *closest* position within the subregion (hyperellipsoid-bounded), however this requires an expensive gradient descent computation, and instead we approximate the subregion as a hypercuboid. The alternative of moving the point directly towards the cluster mean is grossly inaccurate for eccentric hyperellipsoids. Figure 2 illustrates.
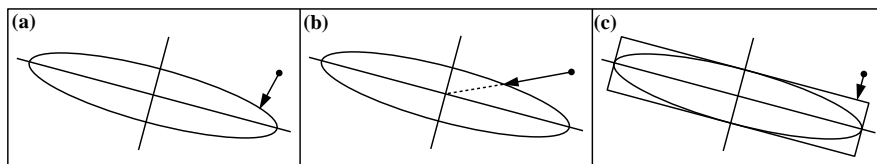

Figure 2: Constraining a general point to lie within a linear subregion; (a) the correct way, (b) a bad approximate method and (c) a better approximate method.

There are other ways to apply the constraints; Bregler describes a constraint function C on a shape $\mathbf{x}$ as follows:

$$C(\mathbf{x}) = \frac{\sum_i G_i(\mathbf{x})P_i(\mathbf{x})}{\sum_i G_i(\mathbf{x})} \qquad (2)$$

where $P_i(\mathbf{x})$ is the shape $\mathbf{x}$ as projected into the subregion of cluster i and G_i is the *influence* function for cluster i. Setting $G_i = 1$ if cluster i is closest and $G_i = 0$ if not results in the simple algorithm described above. Alternatively, G_i can be a Gaussian, centred on cluster i's mean:

$$G_i(\mathbf{x}) = \exp\left(\frac{\|\mathbf{x} - \mu_i\|^2}{\lambda_i^2}\right) \qquad (3)$$

where μ_i is the cluster mean and λ_i is a scale factor related to the 'size' of cluster i. A sensible value for λ_i is the square root of the sum of the eigenvalues from the local PCA. It is important to note that in the above equation, the *Euclidian* distance (as opposed to the Malhalanobis distance) is used; otherwise the influence function decays too quickly off-axis for eccentric hyperellipsoids.

For a particular $\mathbf{x}$, $G_i(\mathbf{x})$ will be very small for the majority of i. The calculation of $C(\mathbf{x})$ can be made more efficient by only including terms for which $G_i(\mathbf{x})$ is significant. A cutoff point of one tenth of the maximum value is suitable.

Using the Gaussian influence functions has the effect of performing an interpolation at positions between neighbouring clusters, giving smoother joins, especially in cases where the subregions don't actually overlap. However, a side effect is that the notion of a concrete divide between valid and invalid shapes is lost, insofar as that if $C(\mathbf{x}) = \mathbf{x}'$ then it is not necessarily the case that $C(\mathbf{x}') = \mathbf{x}'$. It is thus important to only apply the constraint function *once* each time the shape needs constraining. Also this method is slower than the nearest-cluster method.

The method we describe here is based on Bregler and Omohundro's approach, but differs in two aspects. Firstly, they treated the linear patches as lower-dimensional hyperplanes, whereas we prefer to use hyperellipsoid-bounded regions. The hyperplanes method has the undesirable property that it extends the VSR indefinitely at extremities (see Figure 3). Secondly, we have introduced the idea of a hierarchical framework, whereby a global PCA is performed prior to the constraint process. This increases efficiency and also removes some training data noise.

An alternative approach is described in the statistics literature. The VSR can be modelled as a probability density function, approximated as a Gaussian mixture. Instead of using k-means to determine the Gaussians, the EM algorithm [4] can be used with equal, if not better, success.
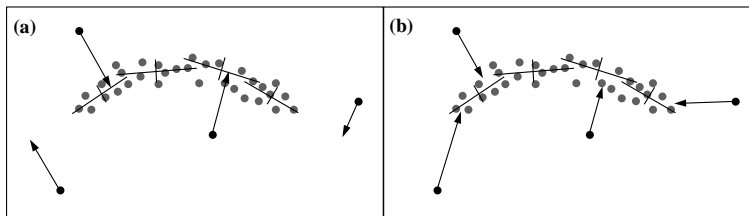

Figure 3: Constraining shape using (a) hyperplanes and (b) hyperellipsoids.

3.1 Choosing k, the n_i and M

k is the number of clusters that are used to build the VSR. n_i is the number of nearest-neighbour training examples used to build the linear subregion for cluster

i. Between them, k and the n_i determine the degree of cluster overlap.

Our current strategy is to specify a *fixed* degree of cluster overlap, O, and set $n_i = Oc_i$, where c_i is the number of members in the k-means cluster. The argument for overlap is that it results in smoother transitions between subregions. Initial experimentation suggests that $O = 1.5$ produces a good balance between accuracy (allowing all valid shapes) and specificity (disallowing invalid shapes).

The choice of k is data-dependent. The more complex and non-linear the VSR, the larger the number of clusters required to model it accurately; however, there is a trade-off between accuracy and speed. If speed is not an issue then k can be increased in the limit to E (but the choice of O must be reconsidered). For noiseless training data this produces the smoothest model; however any noise that *is* present is liable to be included in the model.

So far we have chosen k manually, based on knowledge about the expected shape of the VSR. It seems likely that it would be possible to find a sensible value for k automatically via some optimisation technique. If unsure, a good first guess would be $k = E/10$.

Also important is the choice of M—the Malhalanobis radius of the clusters. We have chosen $M = 2.0$ which, statistically, encompasses over 95% of the cluster members. A larger value generally leads an underconstrained shape space.

4   Evaluation

4.1   Synthetic Data - an Anglepoise Lamp

An anglepoise lamp consists of a fixed base and three rigid jointed sections. This was modelled in 2D using 49 landmarks. Training examples were generated by choosing uniformly-distributed random values for the three pivot angles. A global (linear) PCA was performed. Figure 4 shows the three most significant modes of variation. As can be seen, even along the principal axes, there are several invalid shapes generated.
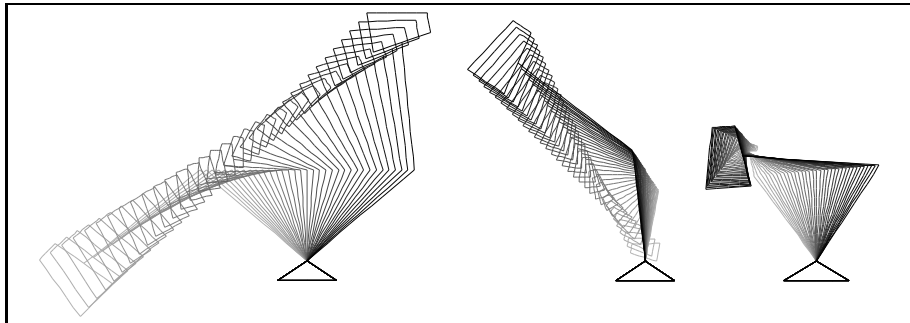

Figure 4: The three most significant modes of variation of the *linear* lamp PDM. Many invalid shapes can be seen.

A hierarchical PDM was then constructed from the same training data. Figure 5 shows the training set with the locally-linear constraint patches superimposed, giving some idea of the VSR that has been learned. The concept of a mode of variation does not exist within the context of a HPDM; the nearest equivalent is to 'drag' the model through shape space, whilst applying shape constraints. Figure 6 illustrates three such drags. The results are much improved over the linear PDM; points are seen to move along arcs, not straight lines, and for the most part the lamp head size remains constant.
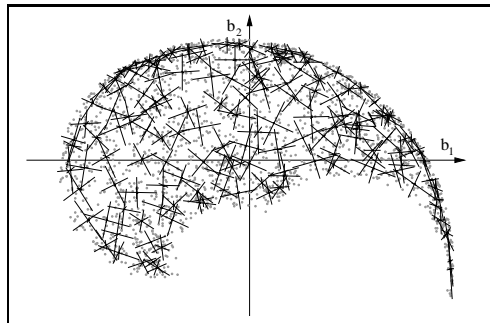
Figure 5: The lamp model shape space (2D projection), showing training data and principal component axes for the constraint regions.
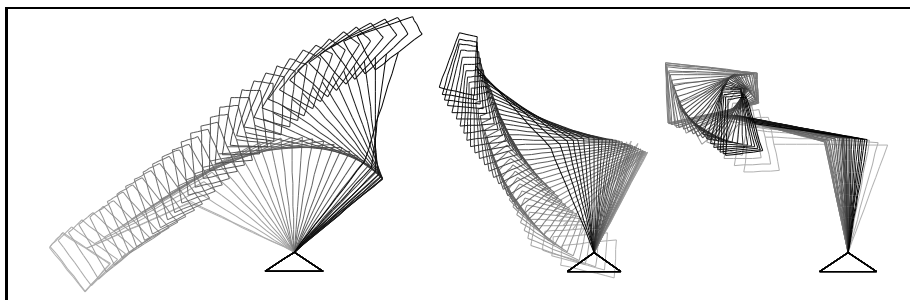

Figure 6: Three constrained 'drags' through shape space for the lamp HPDM.

4.1.1 Specificity

To measure the degree of model specificity (ability to exclude invalid shapes), a large number of shapes were generated, distributed randomly in the shape space. The HPDM constraints were applied, and the distance (in shape space) to the nearest position in the 'ground-truth' VSR was found (approximated as the distance to the nearest of a large number of valid shapes) We have defined the specificity error of the model to be the 90th percentile of these distances (this reflects maximum error whilst excluding outliers). Figure 7 shows the effect on the specificity error of the lamp's HPDM as the number of clusters is varied. The degree of overlap is fixed for each plot, so as the number of clusters increases, the cluster size decreases accordingly. A lower bound to specificity error is experienced due to our approximation of the 'ground-truth' VSR; this is the average distance between nearest-neighbour pairs in the ground truth shape set, and is shown as 'test accuracy' on the graph. Plots for the linear PDM and Bregler's hyperplane approach are also shown for comparison.

As expected, the model specificity error decreases as the number of clusters increases, since the VSR is being approximated more and more accurately with an increasing number of linear patches. The graph levels out at around $k = 20$, at which point the HPDM's specificity error is roughly a quarter of that of the linear PDM. The improvement in specificity achieved by using hyperellipsoid-bounded regions as opposed to hyperplane constraints is also visible, as is the degradation in specificity caused by increasing the degree of overlap—this is because larger linear patches are covering increasingly non-linear subregions of the VSR.
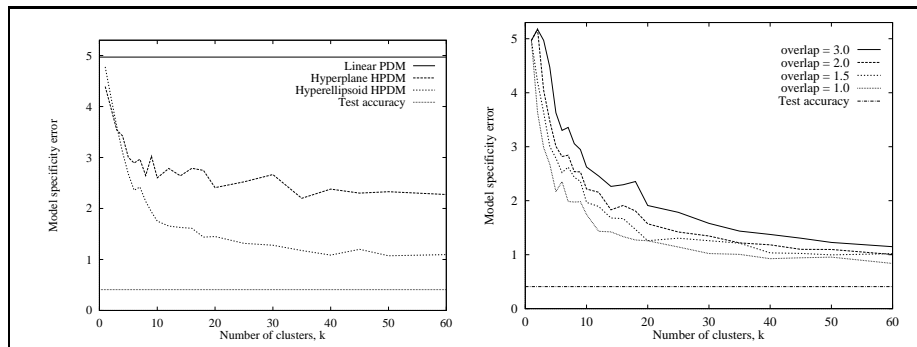
Figure 7: Specificity of the lamp HPDM as the number of clusters is varied, showing (left) a comparison of algorithms and (right) varying degrees of overlap.

4.2 Automatically collected real data

The main motivation for this work was the desire to build models from automatically collected training data. Hand shapes were sampled directly from a live video stream. Various gestures were performed against a black background; the image was thresholded and the hand outline extracted using a simple boundary-finding algorithm. 100 landmarks were positioned at equal intervals around the boundary. This method of data collection suffers greatly from the problem that landmarks rarely mark the same object feature across training examples. For example, when the fingers are outstretched the boundary is much longer than for a pointing gesture; the landmarks spread out more and tend to 'slide' round the boundary. There were 1079 training examples in all; Figure 8 shows some examples.
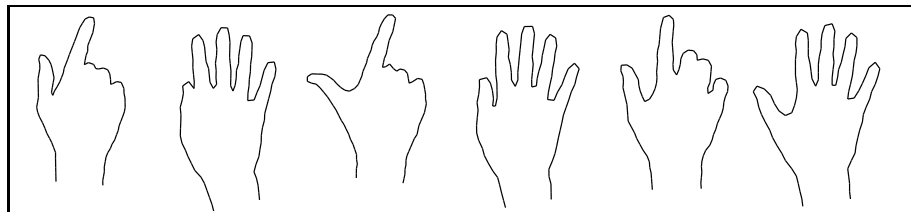

Figure 8: Automatically collected training examples for a hand model.

A HPDM was constructed from the training data, using 80 clusters. Figure 9 shows several views of the training data in the global PCA shape space, along with the clusters found, and Figure 10 shows the four major non-constrained modes of variation (top row) and the four equivalent 'drags' for the HPDM (bottom row).

Figure 9 clearly illustrates that the training data is virtually one-dimensional in nature; representing transitions between the various gestures, however the paths through the shape space are highly non-linear, spiraling through many dimensions.

Figure 10 demonstrates how in this case a Linear PDM fails to produce a model which would be specific enough for object tracking or location. The HPDM 'drags' include only valid object shapes. There appear to be discontinuities in various drags; this is expected because the VSR is not necessarily continuous parallel to any one axis in the global PCA space.
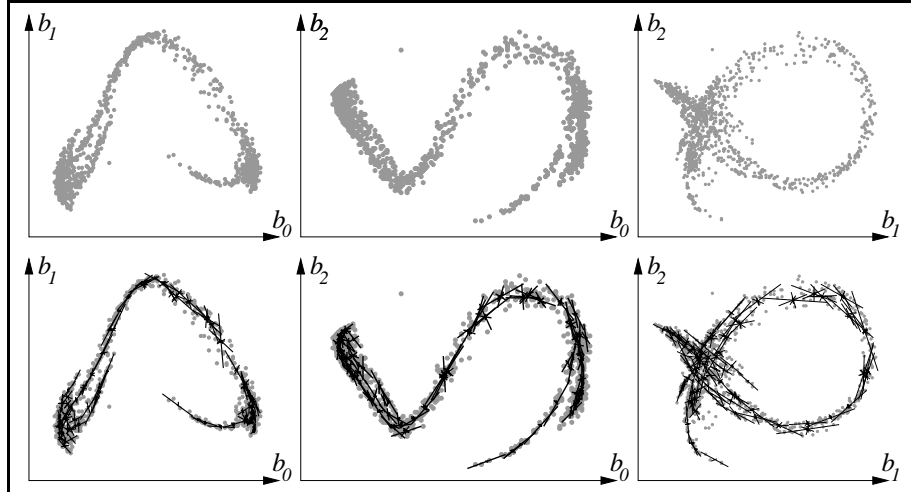
Figure 9: Several views of the automatically collected hand training data in shape space (top row) and the HPDM clusters found (bottom row).

5   Conclusions

We have described the construction of 'hierarchical' PDMs, using a piecewise linear PCA strategy. We have performed both qualitative and quantitative analyses on synthetic data, and also examined performance on automatically-collected real data, with promising results. The HPDM is a viable solution to the problem of fully-automated construction of deformable models.

The hierarchical approach requires a large amount of training data to build good models. However, this problem is negated by the fact that training data can be collected automatically. In the example of the hand we gave, it took less than 5 minutes to collect all the training data and build the model. More intelligent training data collection (eg. Hill and Taylor's approach [7]) might give rise to a less complex shape space which could then be modelled with fewer linear pieces.

Another issue is that of speed. When applying the shape constraints it is necessary to calculate distances to every cluster. This process is order n in the number of clusters. Bregler and Omohundro suggest the use of 'Bumptrees' [8] (a tree-like data structure for representing functions and constraints) to decrease the number of calculations. A related approach would be to extend the hierarchical model to more than two levels, inserting intermediate-sized PCA spaces between the coarsest (global) and finest levels to give a multi-level tree structure. Search for the nearest cluster(s) would descend through the tree, giving at worst order $n \log n$ performance and maybe better in the case of only a partial tree descent.

Tracking using HPDMs has proven relatively successful, proceeding in much the same way as for a standard PDM [3]; HPDMs are less likely to be distracted by image noise and background clutter. However, in the case of automatically trained models, some deformations are not tracked well, specifically those which require landmarks to 'slide' around the model boundary or those which give rise to sudden shape changes. A solution to this problem is discussed in [5].
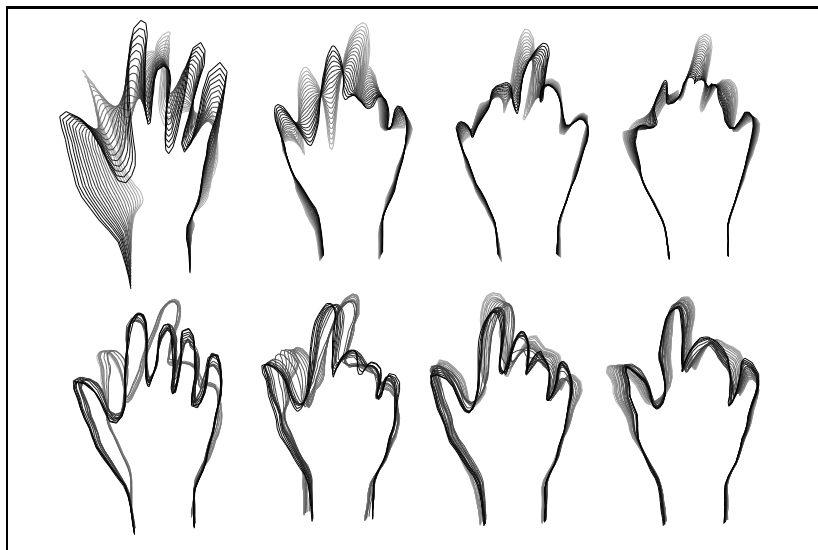
Figure 10: Modes of variation for the automatically trained hand model. Non-constrained modes (top row) and equivalent constrained 'drags' (bottom row).

References

[1] T. Ahmad, C.J. Taylor, A. Lanitis, and T.F. Cootes. Tracking and recognising hand gestures using statistical shape models. In *Proc. BMVC*, pages 403–412, Birmingham, UK, 1995. BMVA Press.

[2] C. Bregler and S. Omohundro. Surface learning with applications to lipreading. In J D Cowan, G Tesauro, and J Alspector, editors, *Advances in neural information processing systems 6*, 1994.

[3] T.F. Cootes, C.J. Taylor, D.H. Cooper, and J. Graham. Active Shape Models - their training and applications. *Computer Vision and Image Understanding*, 61(2), January 1995.

[4] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society*, B39:1–38, 1977.

[5] A.J. Heap and D.C. Hogg. Wormholes in shape space: Tracking through discontinuous changes in shape. (submitted to ICCV '98).

[6] A.J. Heap and D.C. Hogg. Extending the Point Distribution Model using polar coordinates. *Image & Vision Computing*, 14(8):589–599, August 1995.

[7] A. Hill and C.J. Taylor. A method for non-rigid correspondence for automatic landmark identification. In *Proc. BMVC*, pages 323–332, Edinburth, UK, 1996. BMVA Press.

[8] S. Omohundro. Bumptrees for efficient function, constraint, and classification learning. In Touretzky Lippmanm, Moody, editor, *Advances in neural information processing systems 3*, pages 693–699, San Mateo, CA., 1991.

[9] P.D. Sozou, T.F. Cootes, C.J. Taylor, and E.C. Di-Mauro. A non-linear generalisation of PDMs using polynomial regression. In *Proc. BMVC*, volume II, pages 397–406, York, UK, 1994. BMVA Press.

[10] P.D. Sozou, T.F. Cootes, C.J. Taylor, and E.C. Di-Mauro. Non-linear point distribution modelling using a multi-layer perceptron. In *Proc. BMVC*, volume I, pages 107–116, Birmingham, UK, 1995. BMVA Press.