Modelling Object Appearance Using The Grey-Level Surface

T.F.Cootes, C.J.Taylor

Department of Medical Biophysics
University of Manchester
Oxford Road
Manchester M13 9PT
email: bim@wiau.mb.man.ac.uk

Abstract

We describe a new approach to modelling the appearance of structures in grey-level images. We assume that both the shape and grey-levels of the structures can vary from one image to another, and that a number of example images are available for training. A 2-D image can be thought of as a surface in 3 dimensions, with the third dimension being the grey-level intensity at each image point. We can represent the shape of this surface by planting *landmark points* across it. By examining the way such collections of points vary across different examples we can build a statistical model of the shape, which can be used to generate new examples, and to locate examples of the modelled structure in new images. We show examples of these composite appearance models and demonstrate their use in image interpretation.

1 Introduction

In order to accurately locate structures in images we must have good models. Structures of interest can often vary in shape and grey-level appearance because they are flexible, are imaged under different conditions, or natural variations are present. On approach is to use deformable models which can accommodate this variation, preferably in a compact manner. In this paper we address the problem of building models to represent the appearance of such variable image structures an using them in image interpretation. Various approaches to modelling shape and shape variation have been described by Yuille *et al.* [2], Kass *et al.* [1] Hinton, Williams and Revow [3] , Staib and Duncan [4], Pentland and Sclaroff [5], Karaolani *et al.* [6], Nastar and Ayache [7], Grenander *et al.* [8] and Mardia *et al.* [9]. Marchant [10] has shown how Finite Element Methods can be used to add grey-level information to shape models. Grenander and Miller [11] also describe models which represent both the shape and intensity information.

A 2-D image can be thought of as a surface in 3 dimensions, with the third dimension being the grey-level intensity at each image point. The image appears as a mountainous landscape, with each structure in the image represented as a series of mountains and valleys. In order to model a structure, we can model these 3-D shapes. This can be achieved by placing a series of labelled 3-D landmark points across the image surface. Given a set of examples of such structures, each labelled with landmarks, we can find the mean shape from the mean position of each landmark, and find the main ways in which the shape tends to vary by examining the statistics of the point positions. This approach to modelling 3-D shape is an extension of earlier work on 2-D

BMVC 1994 doi:10.5244/C.8.47

shape (Cootes *et al.* [12]) and has been successfully used with 3-D volume data images (Hill *et al.* [13]). In our current work, however, the third dimension is the image intensity, so variations in this dimension cause the grey-levels of the structures to change. We describe the model building approach, give some examples of models and show how they can be used to locate examples in new images.

2 Building Models of the Image Surface

2.1 Representing the training set.

We assume that a number of example images are available containing the structure we wish to model. We will use a model of an eye as a simple example. Figure 1 shows a number of examples of an eye, taken from different images of the same face.

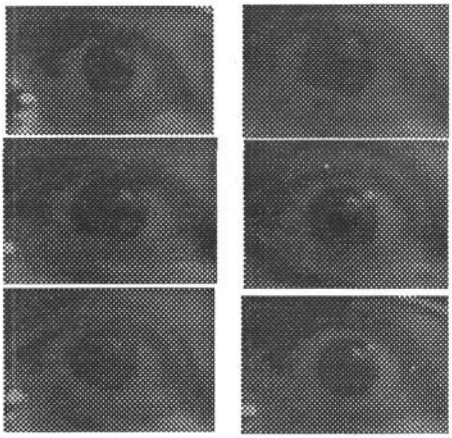

Figure 1 : Examples of eye images

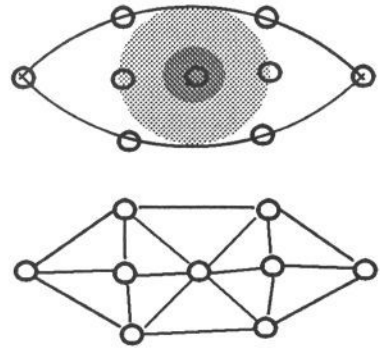

Figure 2 : Base landmark points and triangulation.

On each example we place a number, n_0, of labelled landmark points. Each labelled point represents the position of a particular part of the structure, a corner, a point of high intensity and so on, and must be placed in the same way on each structure (see Cootes *et al.* [12]). We choose the landmarks to represent significant parts of the structure, and to give a reasonably even spread across the surface of the structure to be modelled. Figure 2 shows the placement of landmarks used for the eye.

In order to represent the grey-levels in the internal areas of the structure we usually require a greater density of points. We have generated more internal points by first triangulating the original points, then adding additional points at the mid-point of each connecting arc (Figure 3). A new, denser triangulation can be generated (Figure 4), and the process repeated to obtain any desired density of points. The triangulation algorithm must not add any arcs outside the original boundary (many algorithms force a convex boundary). The algorithm we used is described in Appendix A.

The triangulation algorithm is not robust to movements of the points – small changes in position of the points can lead to a different set of connecting arcs (Figure 5). Since we intend increasing the density of points by adding additional ones along the arcs,

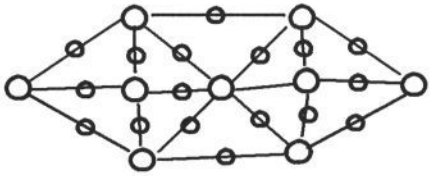

Figure 3 : Interpolated points (one on each connecting arc) on eye example

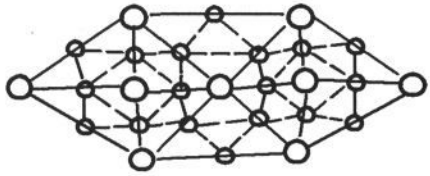

Figure 4 : Denser triangulation for eye example, formed by connecting pairs of new points.

we cannot simply apply the technique to each set of data points independently, as the points generated would no longer be equivalent from one example to another. Instead we only apply the triangulation algorithm to the mean configuration of landmarks. This mean is generated by *aligning* the sets of examples so that they overlap as much as possible , then calculating the mean of each co–ordinate for each landmark point. We record the list of connecting arcs, and the new connecting arcs generated at each level of the triangulation. This data allows us to apply equivalent triangulations to every example, and to generate internal points in a consistent manner.

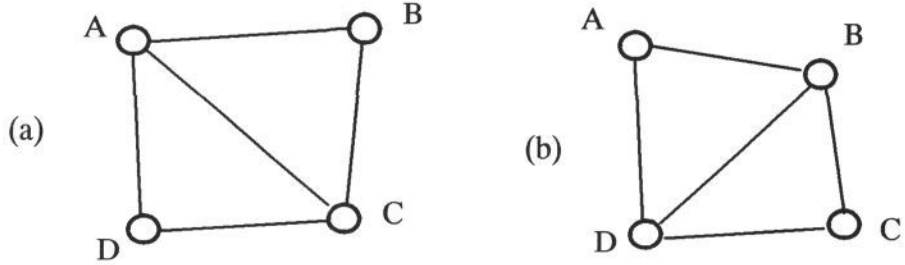

Figure 5 : Example showing instability of triangulation algorithms. In (a) A and C are closer than B and D, so an arc is generated connecting them. In (b) point B has moved a small distance inward, making BD smaller than AC, so a different diagonal is created.

For each example we now have a larger set of n points (x_i, y_i) $(i = 0..n-1)$. Note that they can all be generated from linear combinations of the original n_0 landmark points. This data can give us information about the shape of the structure, but to obtain information about its grey–level appearance we must sample the image at each point to get the grey–level intensity at that point, I_i (Figure 6). Because the points need not have integer co–ordinates, we use bilinear interpolation of the image data to calculate the I_i. Each point becomes a triplet, (x_i, y_i, I_i), which can be considered as a point in a 3–D space. Each set of n points describes the shape of a surface in this 3–D space. We have a set of examples, and can use the 3–D shape model similar to the one described by Hill *et al* [13] to represent their mean and allowable variation.

2.2 Building 3–D Surface Models from Sets of Examples.

The example shapes are *aligned* so that they overlap as much as possible. We allow a rigid rotation, scaling and translation in the x–y plane, and a scaling and translation

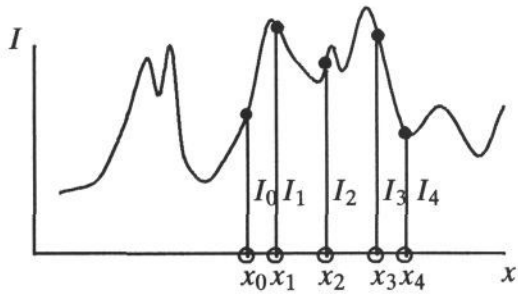

Figure 6 : Intensity information is gathered at each point

in the *I* direction. A least squares technique can be used to align any two sets of labelled points, to minimise the distance between equivalent points on different shapes. An iterative approach can be used to align together a whole set of shapes. Every shape is first aligned with the first shape in the set. The mean is calculated, and this is also aligned with the first shape. Each shape is then re–aligned with the mean, and the mean recalculated and aligned with the first shape. After a few iterations the process converges.

Each structure is then represented by a set of aligned points, (x_i, y_i, I_i), which can be formed into a single $3n$ dimensional vector :

$$\mathbf{x} = (x_0, y_0, \lambda I_0, \; x_1, y_1, \lambda I_1, \; \ldots, \; x_{n-1}, y_{n-1}, \lambda I_{n-1}) \qquad (1)$$

where λ is a proportionality constant to allow for x and y being measured in different units to the grey–level intensity.

We apply a Principal Component Analysis to the set of example vectors. This involves calculating the mean and the covariance matrix about that mean. The eigenvectors of the covariance matrix corresponding to the largest eigenvalues describe the main ways in which the samples vary from the mean. Every example can be approximated using

$$\mathbf{x} = \hat{\mathbf{x}} + \mathbf{Pb} \qquad (2)$$

where $\hat{\mathbf{x}}$ is the mean set of points,

P is a $3n$ x t matrix, the columns of which are the t orthonormal unit eigenvectors of the covariance matrix corresponding to the largest eigenvalues. Each column describes a *mode* of shape variation in the data, the first being the most significant,

b is a set of t model parameters.

By varying the model parameters $\mathbf{b} = (b_1, b_2, \ldots b_t)$ within certain limits which can be learnt from the statistics of the training set, we can generate new examples of the 3–D shape, similar to those in the training set. Varying each parameter causes changes to both the position of each landmark and the intensity value at that landmark.

Given such a 3–D shape **x** we can generate the image structure it represents by plotting I_i at each point (x_i, y_i). If necessary we can interpolate between landmark points on the 3–D surface to fill in internal grey–levels in the 2–D image structure.

3 Examples of Models

3.1 Model of Eye

A model was trained on 10 examples of images of a left eye taken from a series of images of one persons face (Figure 1). Nine landmarks were placed on each image, as shown in Figure 2. The sets of points were aligned and the mean shape calculated. The triangulation algorithm was applied, and points were interpolated along each arc. This was repeated twice to get a total of 345 points. Equivalent sets of points were interpolated for each of the 10 examples. At every point in each example the intensity was sampled (using bilinear interpolation) and a model was built from this data. The relative importance of intensity to x,y values, λ, was set at 0.25.

Figure 7 shows the mean appearance. Figure 8 shows the effects of varying each of the first four model parameters by ± 4.0 standard deviations of their values across the training set. Changing the parameters varies both the shape of the model and the intensities across its area (though the effect is subtle in this example). For instance the first parameter varies the overall brightness of the iris, and controls the intensity of the highlight in the upper right area. The third parameter varies the shape of the model, with only small changes in intensity. The examples in the original training set are all quite similar, so there is little variation in the model.

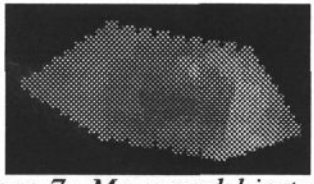

Figure 7 : Mean model instance

3.2 Model of Banana

In order to examine the ability of the model to capture both shape changes and variations in grey-levels due to changes in lighting conditions, several images were taken of a set of bananas, illuminated by spotlights from different directions (Figure 9). The bananas were selected from several bunches (though all were from the same plantation). 33 points were marked on each banana in each training image. A triangulation was applied to the mean shape, and two iterations of the interpolation procedure were used on each example to produce 369 points. A model was trained on these, with the relative importance of intensity to x,y values, $\lambda = 0.25$. Figures 10 to 12 show the effects of varying the first 5 model parameters between ± 4.0 standard deviations of their values across the training set. Mode 1 (Figure 10) mainly affects the shape of the model. Other modes combine small shape variations with shading variations. Mode 2 (Figure 11) shows the effect of the illumination coming from different sides and mode 3 (Figure 12) shows a general lightening and darkening.

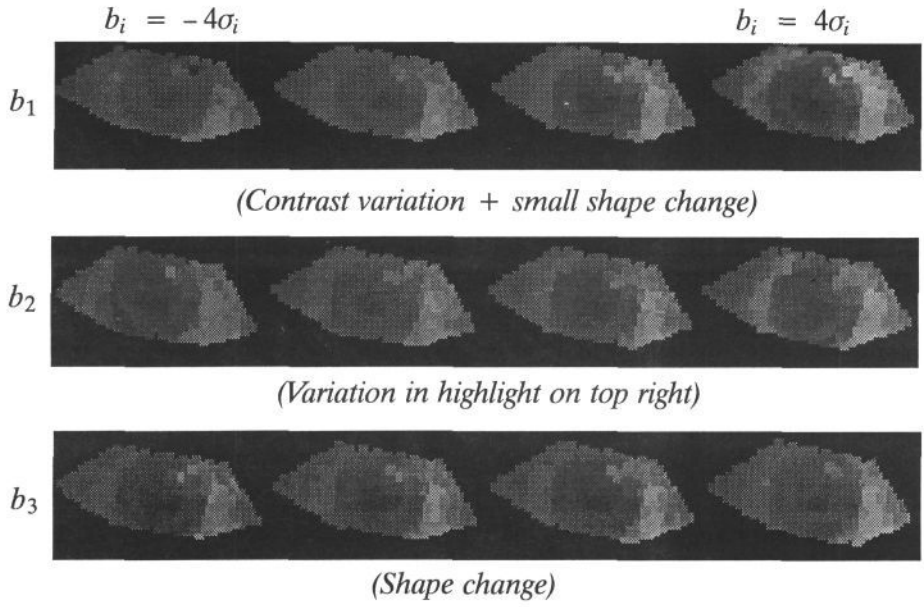

Figure 8 : Effect of varying each of first four model parameters of eye model within ±4 standard deviations (other parameters being set to zero).

Figure 9 : Some of the bananas used to train model.

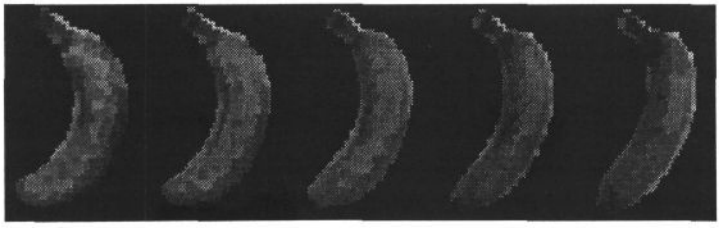

Figure 10 : Effect of varying the first model parameter within ±4.0 s.d.s. (All others 0) (Shape and overall intensity change)

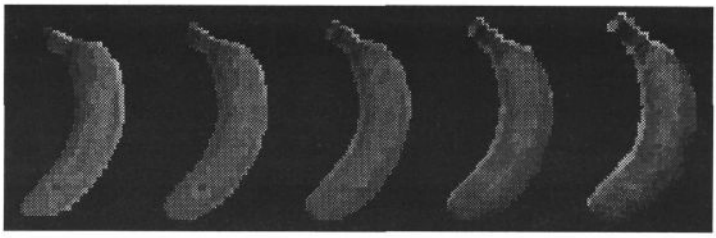

Figure 11 : Effect of varying the second model parameter within ±4.0 s.d.s. (All others 0) (Illumination angle change)

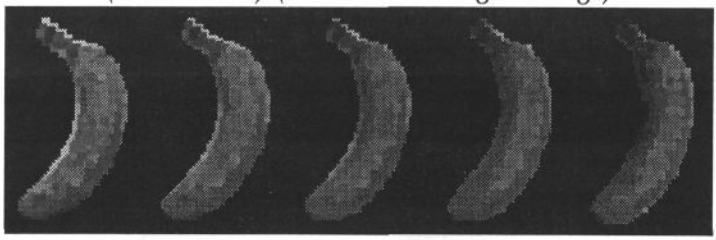

Figure 12 : Effect of varying the third model parameter within ±4.0 s.d.s. (All others 0) (Shape change)

4 Using Models in Image Search

Image search usually involves finding the translation, rotation, scale and model shape parameters required to give the best fit between a model instance and the target structure in the image. This requires a definition of the 'quality of fit'.

Suppose the current instance of the model in its local co–ordinate frame is given by **x** (generated from the shape parameters, **b**, using Eq. 2). The the positions of the points in the image frame is given by

$$\mathbf{X} = M(s, \theta, s_z)[\mathbf{x}] + \mathbf{X}_c \qquad (3)$$

where $\mathbf{X}_c = (X_c, Y_c, \lambda I_c, \ldots, X_c, Y_c, \lambda I_c)^T$

$M(s, \theta, s_z)[.]$ performs a rotation by θ, a scaling in x–y by s and a scaling in I by s_z.

$(X_c, Y_c, \lambda I_c)$ is the position of the centre of the model in the image frame.

If the image intensity at (x,y) is $I(x,y)$ then we can measure the quality of fit of a model instance using

$$F = \frac{1}{n} \sum_{i=0}^{i=n-1} (I(x_i, y_i) - I_i)^2 \qquad (4)$$

By minimising F we find the best fit of the model to the image.

4.1 Global Search using a Genetic Algorithm

One approach to this minimisation is to use a Genetic Algorithm (GA) to search the parameter space $(X_c, Y_c, I_c, \theta, s, s_z, b_1, \ldots, b_t)$. The GA controls a set of many possible solutions, and combines pairs of promising ones to improve the overall quality. After

a number of *generations* the solutions converge to an estimate of the global optimum solution [13].

We have implemented a version of the GA to choose the model parameters which minimise F for a given image, hopefully locating instances of the model in the image. For instance Figure 13 shows the location of the best fitting model found using the Genetic Algorithm to search around the vicinity of the eye in a new image (A population of 50 was used, and about 50 generations were required for the results shown). Figure 14 shows the appearance of this best fitting model. It demonstrates that as well as locating the position of the eye, the model has fitted itself to the intensity pattern, for instance showing the highlight in the upper right area of the iris.

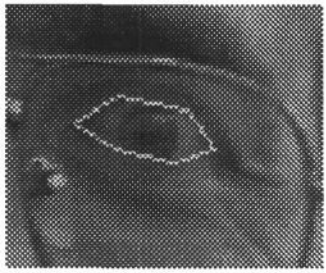

Figure 13 : Location of model found by Genetic Algorithm search.

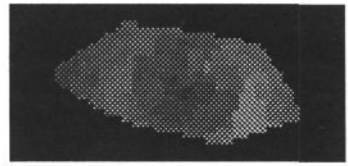

Figure 14 : Appearance of best fitting model.

5 Discussion and Conclusions

The method we describe gives a way of generating a flexible model which combines both the shape and grey-level appearance of structures in an image. The model records both the mean appearance of the structure and the ways its shape and grey-levels can vary, based on the original training examples. Although not clearly demonstrated above, we believe it will be able to model the way that variations in shape cause variations in shading.

An alternative approach to modelling shape and grey-level appearance is to use independent models for each. Craw and Cameron [14] and Lanitis *et al* [15] both first locate a set of landmarks in an image, then deform the image so that the landmarks move to their mean model positions. Statistical models are built from the grey-levels in important areas of the deformed images. In this approach the independence of the shape of the object from the grey-levels means that relationships between shading and shape may not be effectively captured.

The choice of the parameter relating grey-level values to x-y coordinates, λ, is not easy. Large values of λ give priority to variations in grey-level appearance - the first modes will be predominantly intensity variations. Small values give priority to shape variations - the first modes will vary shape. The choice of the most suitable value will depend on the nature of the application. If discrimination between different objects is required, λ should be set so as to bring out the most discriminating modes of variation, whether they be predominantly shape or intensity based. It should be possible to choose a value for λ depending upon the amount of variation in the training set.

We have demonstrated that the models can be used to locate structures in images using a Genetic Algorithm to control the search. Since the new models include more information than our earlier models representing the shape and grey-levels in regions around the boundary [12], it is hoped that they may provide a more robust estimate of object location.

Acknowledgements

Tim Cootes is funded by an SERC Postdoctoral Fellowship. The authors would like to thank the other members of the Wolfson Image Analysis Unit for their help and advice.

Appendix A : Triangulation Algorithm

We require a triangulation of a set of points, with the restriction that the connecting arcs should not stray outside a (possibly concave) boundary. Standard techniques (eg Delaunay triangulation) will fill in concave sections to produce a convex hull.

We implemented a variation of the *greedy* triangulation algorithm. This is not the most efficient approach, but is relatively simple, and in our problem speed is not too important, since the triangulation is only done once, during the training stage.

We assume that we have a set of n points, and that the boundary of the set of points is defined by a series of arcs, marked clockwise between the external points (Figure 15).

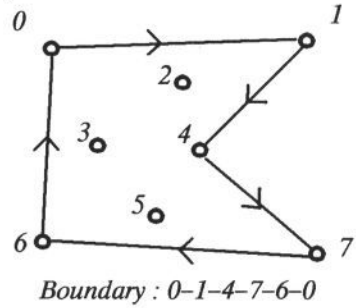

Figure 15 : A set of points with boundary marked clockwise.

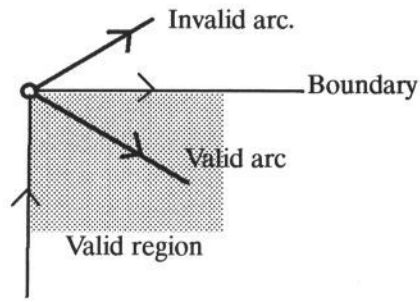

Figure 16 : Any arc to be added at a boundary must lie inside the boundary.

We first calculate the lengths of the $n(n-1)/2$ possible connecting arcs between the points, r_{ij}. This list is sorted into ascending order. The arcs defined in the boundary are recorded as the initial set of arcs. Each arc in the sorted list, starting with the shortest, is then examined to see if it is suitable for adding to this set of arcs. It is accepted if

- It intersects none of the current set of arcs, except at a point.
- It does not lie outside the boundary.

The second criterion need only be checked if both ends of the arc are boundary points. If one is not, the arc must cross a boundary arc and will fail the first test. To test whether an arc lies inside the boundary, one need only check that its angle is inside a range defined by the two boundary arcs at one of the end points (Figure 16).

The algorithm terminates when all possible arcs have been considered. The set of arcs which were accepted defines the triangulation.

References

[1] M. Kass, A. Witkin and D. Terzopoulos , Snakes: Active Contour Models, in *Proc. First International Conference on Computer Vision*, pp 259–268 IEEE Computer Society Press, 1987.

[2] A.L. Yuille, P. Hallinan and D.S. Cohen Feature extraction from faces using deformable templates, IJCV, **8**, August 1992, pp. 99–112.

[3] G.E.Hinton, C.K.I. Williams and M.D. Revow, Adaptive Elastic Models for Hand–Printed Character Recognition. in *Advances in Neural Information Processing Systems 4,* (J.E.Moody, S.J.Hanson, R.P.Lippmann. Ed.s) Morgan Kauffmann, San Mateo, CA, 1992.

[4] L.H. Staib and J.S. Duncan, Parametrically Deformable Contour Models, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition. San Diego, 1989*, pp. 427–430.

[5] A. Pentland and S. Sclaroff, Closed–Form Solutions for Physically Based Modelling and Recognition, *IEEE Trans. on Pattern Analysis and Machine Intelligence*. **13**, 1991, 715–729.

[6] P. Karaolani, G.D. Sullivan, K.D. Baker and M.J. Baines, A Finite Element Method for Deformable Models. *Proceedings of the Fifth Alvey Vision Conference, Reading, 1989*, pp. 73–78.

[7] C. Nastar and N. Ayache, Non–Rigid Motion Analysis in Medical Images : a Physically Based Approach. in *Proceedings of IPMI '93*, (H.H.Barrett, A.F.Gmitro. Ed.s) Springer–Verlag, Berlin 1993, pp.17–32.

[8] U. Grenander, Y. Chow and D.M. Keenan, *Hands. A Pattern Theoretic Study of Biological Shapes*. Springer–Verlag, New York, 1991.

[9] K.V. Mardia, J.T. Kent and A.N. Walder, Statistical Shape Models in Image Analysis, *Proceedings of the 23rd Symposium on the Interface, Seattle 1991*, pp 550–557.

[10] J.A. Marchant, Adding Grey Level Information to Point Distribution Models Using Finite Elements. in *Proc. British Machine Vision Conference.* Ed. J. Illingworth, BMVA Press, 1993, pp.309–318.

[11] U. Grenander, M.I.Miller, Representations of Knowledge in Complex Systems. *J.R.Statistical Soc.:B* (1994) **56**, No.3.

[12] T.F.Cootes, A.Hill, C.J.Taylor, J.Haslam, The Use of Active Shape Models for Locating Structures in Medical Images. *Image and Vision Computing* Vol.12, No.6, July 1994, pp 355–366.

[13] A. Hill, A. Thornham and C.J.Taylor, Model–Based Interpretation of 3D Medical Images. in *Proc. British Machine Vision Conference.* Ed. J. Illingworth, BMVA Press, 1993, pp. 339–348.

[14] I. Craw and P. Cameron. Face Recognition by Computer. *Proc. British Machine Vision Conference. 1992* Eds. D.Hogg, R.Boyle, Springer–Verlag, 1992. pp 489–507.

[15] A. Lanitis, C.J.Taylor, T.F.Cootes, An Automatic Face Identification System Using Flexible Appearance Models. *BMVC 1994*