Active Shape Models—Their Training and Application

T. F. COOTES, C. J. TAYLOR, D. H. COOPER, AND J. GRAHAM*

Department of Medical Biophysics, University of Manchester, Oxford Road, Manchester M13 9PT, England

Model-based vision is firmly established as a robust approach to recognizing and locating known rigid objects in the presence of noise, clutter, and occlusion. It is more problematic to apply model-based methods to images of objects whose appearance *can* vary, though a number of approaches based on the use of flexible templates have been proposed. The problem with existing methods is that they sacrifice model specificity in order to accommodate variability, thereby compromising robustness during image interpretation. We argue that a model should only be able to deform in ways characteristic of the class of objects it represents. We describe a method for building models by learning patterns of variability from a training set of correctly annotated images. These models *can* be used for image search in an iterative refinement algorithm analogous to that employed by Active Contour Models (Snakes). The key difference is that our *Active Shape Models can* only deform to fit the data in ways consistent with the training set. We show several practical examples where we have built such models and used them to locate partially occluded objects in noisy, cluttered images. © 1995 Academic Press, Inc.

1. INTRODUCTION

We address the problem of locating examples of known objects in images. Image interpretation using rigid models is well established [1, 2]. However, in many practical situations objects of the same class are not identical and rigid models are inappropriate. In medical applications, for instance, the shape of organs can vary considerably through time and between individuals. In addition, many industrial applications involve assemblies with moving parts, or components whose appearance can vary. In such cases flexible models, or deformable templates, can be used to allow for some degree of variability in the shape of the imaged objects [3–23].

In this paper we present new methods of building and using flexible models of image structures whose shape *can* vary. The models are able to capture the natural variability within a class of shapes and *can* be used in image search to find examples of the structures that they represent. Previous approaches have allowed models to deform, but have not tailored the Variability to the class of shapes concerned—the models are not specific. Our main contribution is to describe how to create models which allow for considerable variability but are still specific to the class of structures they represent.

Our technique relies upon each object or image structure being represented by a set of points. The points can represent the boundary, internal features, or even external ones, such as the center of a concave section of boundary. Points are placed in the same way on each of a training set of examples of the object. This is done manually, though tools are available to aid the user. The sets of points are aligned automatically to minimize the variance in distance between equivalent points. By examining the statistics of the positions of the labeled points a "Point Distribution Model" is derived. The model gives the average positions of the points, and has a number of parameters which control the main modes of variation found in the training set.

Given such a model and an image containing an example of the object modeled, image interpretation involves choosing values for each of the parameters so as to find the best fit of the model to the image. We describe a technique which allows an initial very rough guess for the best shape, orientation, scale, and position to be refined by comparing the hypothesized model instance with image data, and using differences between model and image to deform the shape. We have previously described how to obtain the initial guess [7]. The method has similarities with the Active Contour Models (or snakes) of Kass et al. [3], but differs in that global shape constraints are applied; to make this distinction clear we have adopted the term Active Shape Models. The key point is that instances of the models can only deform in ways found in the training set.

Our results demonstrate that the method for constructing models combined with the active matching technique provides a systematic and effective paradigm for the interpretation of complex images. In the remainder of the paper we review some of the relevant literature, describe the modeling method, and show examples of

* E-mail: bim@uk.ac.man.mb.wiau. Fax: 061 275 5145.

trained models. The active matching technique is described and results are given, showing how the models can be used to interpret images.

2. BACKGROUND

There is a substantial literature describing the use of flexible models or deformable templates to aid image interpretation. Such models usually have a number of parameters to control the shape and pose of all or parts of the model. We give a brief review of some of the most significant work, which relates mainly to two-dimensional images.

2.1. "Hand Crafted" Models

Flexible models can be built up from simple subcomponents, such as circles, lines, or arcs, which are allowed some degree of freedom to move around relative to one another, and possibly change scale and orientation. Yuille *et al.* [5] model parts of the face, such as the eyes and mouth, in this way. When attempting to fit a model to an image they first obtain an approximate fit, which they refine by changing different parts of the model, one at a time. Lipson *et al.* [6] apply a similar scheme to map elliptical models of vertebrae onto CT images of the spine. Hill *et al.* [7] use a handcrafted model of the heart in combination with Genetic Algorithm search to find the left ventricle in echocardiograms.

Although such models can capture detailed knowledge of expected shapes, the approach lacks generality. It is necessary to design both a new model and a scheme for fitting to images for each application.

2.2. Articulated Models

A number of authors consider articulated models built from rigid components connected by sliding or rotating joints. Beinglass and Wolfson [8] describe a scheme for locating such objects using a Generalized Hough Transform with the point of articulation as the reference point for each subpart. Connected subparts then vote for the same reference point. Grimson [2] has extended his "interpretation tree" approach to object recognition to include some articulations, and reviews other work along the same lines. This approach is only applicable to a restricted class of variable shape problems.

2.3. Active Contour Models ("Snakes")

Kass *et al.* [3] describe flexible contour models which are attracted to image features. These energy minimizing spline curves are modeled as having stiffness and elasticity and are attracted toward features such as lines and edges. Constraints can be applied to ensure that they remain smooth and to limit the degree to which they can be bent.

Snakes can be considered as parameterized models, the parameters being the spline control points. They are usually free to take almost any smooth boundary with few constraints on their overall shapes. The idea of fitting by using image evidence to apply forces to the model and minimizing an energy function is effective.

Hinton *et al.* [4] describe a type of spline snake governed by a number of control points which have preferred "home" locations to give the snake a particular default shape. Deformations are caused by moving the control points away from their "home" locations. Although the average shape of an object is represented, the modes of shape variation are only coarsely defined by the number and position of control points.

2.4. Fourier Series Shape Models

Scott [9] proposes a method of modeling shapes by an expansion of trigonometric functions,

$$x = x_0 + \sum_n a, \sin(n\theta + \phi_n)$$
$$y = y_0 + \sum_n b_n \sin(n\theta + \psi_n). \tag{1}$$

The shape produced is a function of the parameters $a,$, $b,$, ϕ_n, ψ_n. By varying the parameters and the number of terms used, different shapes can be generated. Scott shows how to fit such a shape model to image data by varying the parameters so as to minimize an energy term. The model is almost infinitely deformable, and contains no prior shape information. Staib and Duncan [10] describe similar Fourier models, and use them to interpret medical images. They derive distributions for each of the parameters over a training set and while fitting the model to an image maximize a probability measure determining how likely it is that the current example is the desired object. Bozma and Duncan [11] describe how such a technique can be used to model organs in medical images. A given shape is represented by a list of values for the parameters and is deformed by varying the parameters from these values. They describe ways of incorporating relationships between several flexible objects by applying constraints to the parameters of the models.

Trigonometric basis functions are not suitable for describing general shapes; for example, using a finite number of terms, they can only approximate a square corner. The relationship between variations in shape and variations in the parameters of the trigonometric expansion is not straightforward.

2.5. Statistical Models of Shape

A number of workers have studied the distributions of sets of "landmark" points which mark significant positions on an object. Goodall [14] discusses the registration

FIG. 2. Image of printed circuit board showing examples of resistors.

3. POINT DISTRIBUTION MODELS

Suppose we wish to derive a model to represent the shapes of resistors as they appear on printed circuit boards, as shown in Fig. 2. Different examples of resistor have sufficiently different shapes *so* that a rigid model would not be appropriate. Figure 3 shows some examples of resistor boundaries which were obtained from backlit images of individual resistors. Our aim is to build a model wh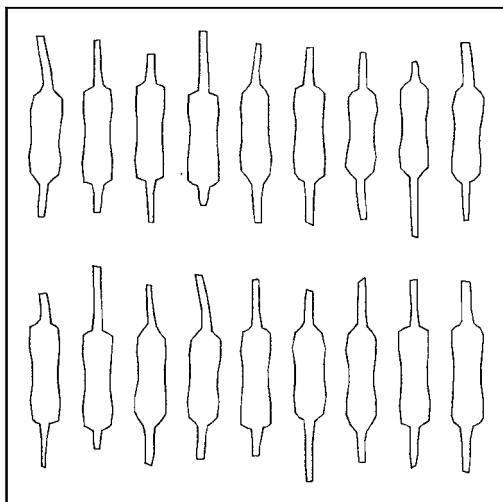ich describes both typical shape and typical variability, using the examples in Fig. 3 as a training set. We achieve this by representing each example as a set of labeled 'landmark' points, calculating the mean positions of the points and the main ways in which the points from each example tend to vary from the mean.

3.1. *Labeling the Training Set*

In order to model a shape, we represent it by a set of points. For the resistors we have chosen to place points around the boundary, as shown in Fig. 4. This must be done for each shape in the training set. The labeling of the points is important. Each labeled point represents a particular part of the object or its boundary. For instance, in the resistor model, points 0 and 31 always represent the ends of a wire, points **3, 4,** and 5 represent one end of the body of the resistor, and *so* on. The method works by modeling how different labeled points tend to move together as the shape varies. If the labeling is incorrect, with a particular point placed at different sites on each training shape, the method will fail to capture shape variability reliably. In the examples shown below the points were either placed manually on each image, or tools were used to mark points on boundaries segmented by hand. It is worth noting that the points are only placed manually during the training phase; it is *not* necessary to find these points in advance when the models are used for image interpretation — we describe later how this is achieved implicitly using an automatic method.

Bookstein [16, 17] labeled significant points in images of biological and medical specimens in order to examine and measure shape changes which could be correlated with other factors. We use representative points to capture shape constraints and build models which may be used to construct plausible new examples of the shape for use in image interpretation. Bookstein calls his representative points ''landmark points'' and describes them in terms of their usefulness. For our purposes they can be reduced to three different types:

1. points marking parts of the object with particular application-dependent significance, such as the center of an eye in the model of a face or sharp corners of a boundary;

2. points marking application-independent things, such as the highest point on an object in a particular orientation, or curvature extrema;

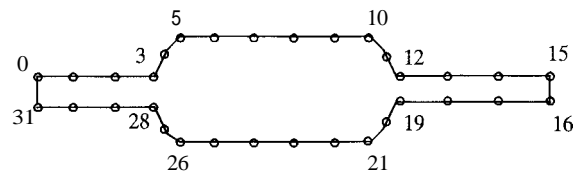FIG. 3. Examples of resistor shapes from a training set.

FIG. 4. Thirty-two point model of the boundary of a resistor.

that of the mean. In this case the landmark point positions will be chosen to best match the mean, rather than rigidly imposed. This leads to better models.

The convergence condition in the alignment procedure can be tested by examining the average difference between the transformations required to align each shape to the recalculated mean and the identity transformation. Experiments show that the method converges to the same result independent of which shape is aligned to in the first stage, though a formal proof of convergence has yet to be devised. We have considered direct methods of solution but have found problems with numerical stability. Since computational efficiency is not an issue during model construction the iterative method is adequate for our purposes.

3.3. Capturing the Statistics of a Set of Aligned Shapes

In Fig. **5** the coordinates of the some of the vertices of the aligned resistor shapes are plotted, with the mean shape overlaid. It can be seen that some of the vertices show little variability over the training set, while others form more diffuse "clouds." The Point Distribution Model (PDM) seeks to model the variation of the coordinates within these clouds. However, it must be remembered that landmarks do not move about independently — their positions are partially correlated.

Each example in the training set, when aligned, can be represented by a single point in a 2n dimensional space (see Eq. **(2))**. Thus a set of N example shapes gives a cloud of N points in this 2n dimensional space. We assume that these points lie within some region of the space, which we call the "Allowable Shape Domain," and that the points give an indication of the shape and size of this region. Every 2n-D point within this domain gives a set of landmarks whose shape is broadly similar to that of those in the original training set. Thus by moving about the Allowable Shape Domain we can generate new shapes in a systematic way. The approach given below attempts to model the shape of this cloud in a high dimensional space, and hence to capture the relationships between the positions of the individual landmark points. We make the assumption that the cloud is approximately ellipsoidal, and proceed to calculate its center (giving a mean shape) and its major axes, which give a way of moving around the cloud. Later we will discuss the implications of this ellipsoid assumption breaking down.

Given a set of N aligned shapes, the mean shape, $\bar{\mathbf{x}}$ (the center of the ellipsoidal Allowable Shape Domain), is calculated using

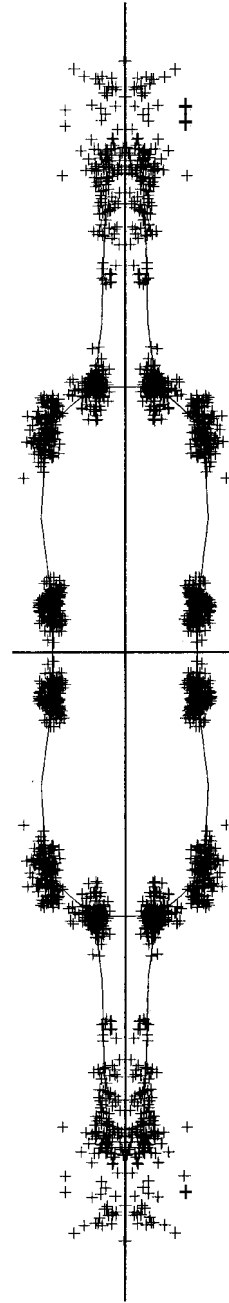$$\bar{\mathbf{x}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_i \qquad (7)$$

FIG. 5. Scatter of some points from aligned set of resistor shapes, with the mean shape overlaid.

The principal axes of a 2n-D ellipsoid fitted to the data can be calculated by applying a principal component analysis (PCA) to the data [25]. Each axis gives a "mode of variation," a way in which the landmark points tend to move together as the shape varies. For each shape in the training set we calculate its deviation from the mean, $d\mathbf{x}_i$, where

$$d\mathbf{x}_i = \mathbf{x}_i - \bar{\mathbf{x}}. \qquad (8)$$

TABLE 1
Eigenvalues of the Covariance Matrix Derived from a Set of Resistor Shapes

Eigenvalue	$\dfrac{\lambda_i}{\lambda_T} \times 100\%$
λ_1	66%
λ_2	8%
λ_3	5%
λ_4	4%
λ_5	3%
λ_6	3%

scription of the deformation. Compare Figs. 7–9 with Fig. **3.** Varying the first parameter *(b,)*adjusts the position of the body of the resistor up and down the wire. The second parameter varies the shape of the ends of the main body of the resistor, between tapered and square. The third parameter affects the curvature of the wires at either end. Subsequent parameters have smaller effects, including the wires bending in opposite directions. These modes of variation effectively capture the variability present in the training set. Note that the apparently large variability in the positions of individual points in Fig. **3** is in fact highly constrained, and the overall variation in shape can be described by a small number of modes. This model has been used to locate resistors in images (see below).

3.4.2. *Heart Model.* Figure 10 shows examples from a set of 66 heart chamber boundaries obtained by asking a cardiologist to draw over echocardiogram images. Each
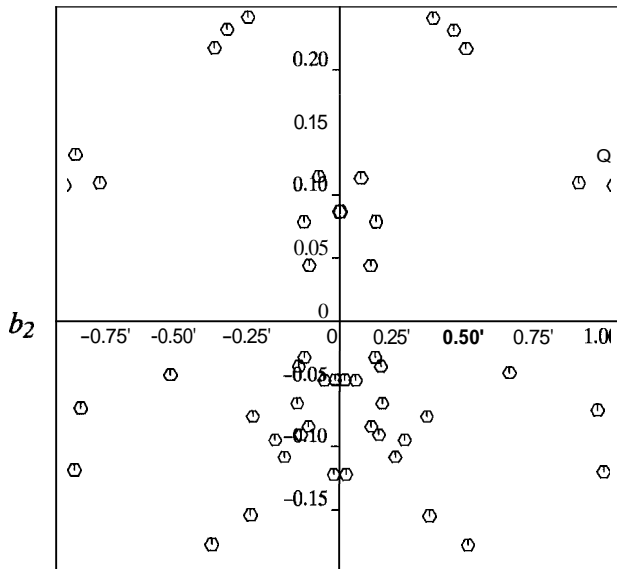
FIG. 6. Plot of b_1 vs b_2 for a training set of resistor shapes.
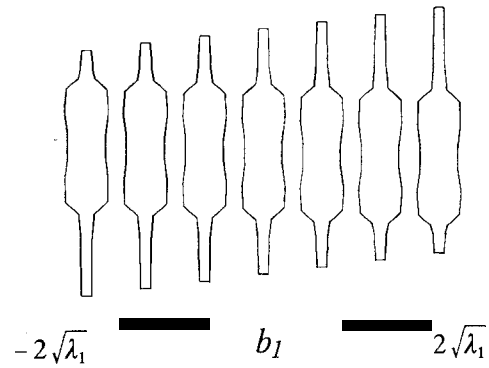
FIG. 7. Effects of **varying the first parameter** of **the resistor model.**

structure is represented by 96 points. This example shows how a single model can represent several shapes *and* the spatial relationships between them. The shape variation arises from two sources: the training set was derived from several individuals, and in each case images were taken from different stages in the cardiac cycle, during which the sizes and shapes of the heart chambers can change considerably. The points represent the boundary of the left ventricle, part of the boundary of the right ventricle, and part of the boundary of the left atrium (below the ventricle in the figures). Table 2 shows the eigenvalues of the covariance matrix obtained for the training set. Figure 11 suggests that b_1 and b_2 are again independent, and Fig. 12 shows reconstructed shapes obtained by varying the first four model parameters in turn. The first parameter varies the width of the shape. The second parameter varies the appearance of the septum (the wall separating the left from the right ventricle). The third and fourth parameters vary the shape of the left ventricle and the modeled part of the atrium below. It should be emphasized that these modes are derived entirely automatically, and arise from a statistical analysis of the variation in the data. This model has been used to locate the boundary of a
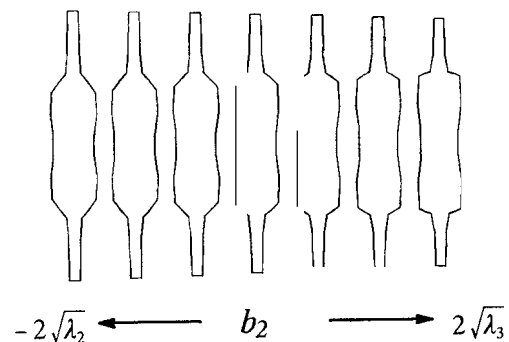
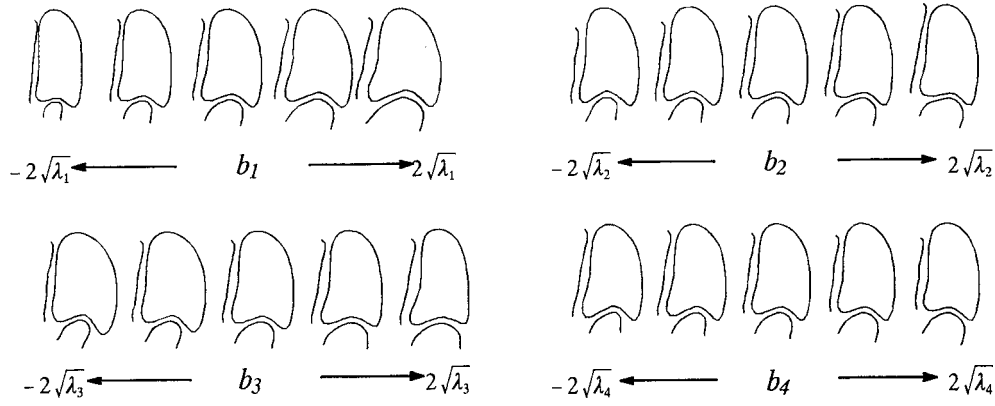**FIG. 8.** Effects of varying the second parameter of the resistor model.

FIG. 12. Effects of varying each of the first four parameters of the heart ventricle model individually.

0 and **6,** form curved clouds, the centroids of which do not lie inside the clouds. The mean shape generated in this way is thus not sufficiently similar to the training set to give a satisfactory model. The first three modes of variation of a PDM trained on this data are shown in Fig. 18. Ideally one would expect a model to have the first and second order curvature as its first two modes. The first mode of the PDM is an approximation to bending, generated by fitting straight lines to the curved "clouds" of points. The second mode gives the corrections required because the linear approximation is poor. The third mode of the model gives an approximation to second order bending. Figure 19 shows the relationship between the first two parameters b_1 and b_2. Though they are *linearly* independent, there are clearly nonlinear relationships present. One cannot choose the parameters independently and ex-
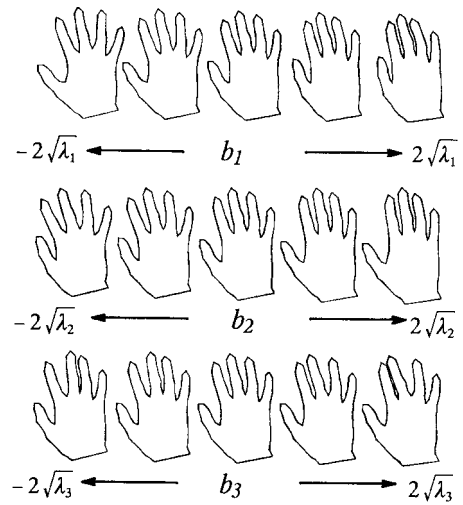

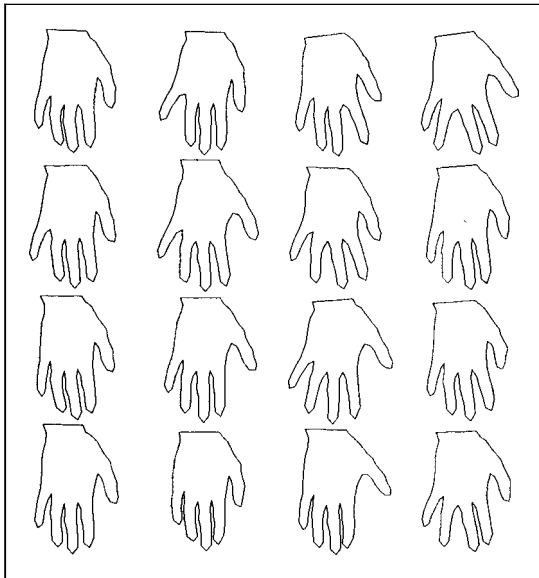**FIG. 14.** Effects of varying each of the first three parameters of the hand model individually.

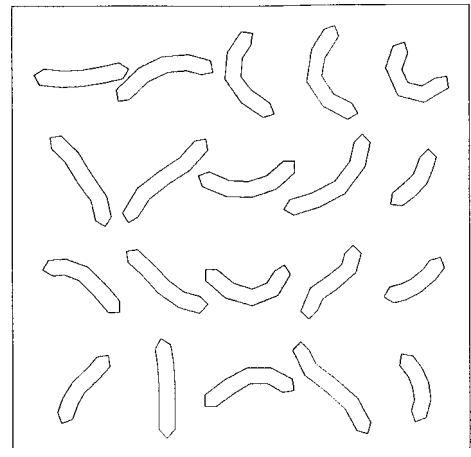**FIG. 13.** Training set of hand shapes, each defined by **72** points.

FIG. 15. Examples from a set of "worm" shapes.

the structures of interest in the image. An instance of the model is given by

$$X = M(s, \theta)[\mathbf{x}] + X,, \quad \text{where}$$

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

$M(s, \theta)[\ ]$ is a rotation by θ and a scaling by s, and   (17)

(X_c, Y_c) is the position of the centre

of the model in the image frame.

In this section we describe an iterative method for finding the appropriate X given a very rough starting approximation. Hill *et al.* have described elsewhere how Genetic Algorithm search can be used to find a good starting approximation quite rapidly [26, 7, 27]; this is applicable if there is no prior knowledge of the expected location of objects of interest. In practice, the starting value of X does not need to be very close to the final solution, so that, for many practical applications, the method below can be used on its own.

The idea of the iterative scheme is to place the current estimate of X into the image and examine a region of the image around each model point to determine a displacement which moves it to a better location. These local deformations are transformed into adjustments to the pose, scale, and shape parameters of the PDM. By enforcing limits on the shape parameters, global shape constraints can be applied ensuring the shape of the model example remains similar to those of the training set. The procedure is repeated until no significant changes result. Because the models attempt to deform to better fit the data, but only in ways which are consistent with the shapes found in the training set, we call them "Active Shape Models" or "Smart Snakes."

4.1. *Calculating a Suggested Movement for Each Model Point*

Given an initial estimate of the positions of a set of model points which we are attempting to fit to an image
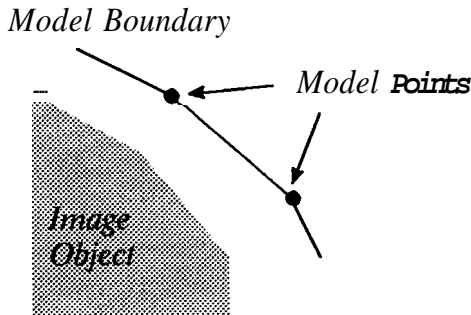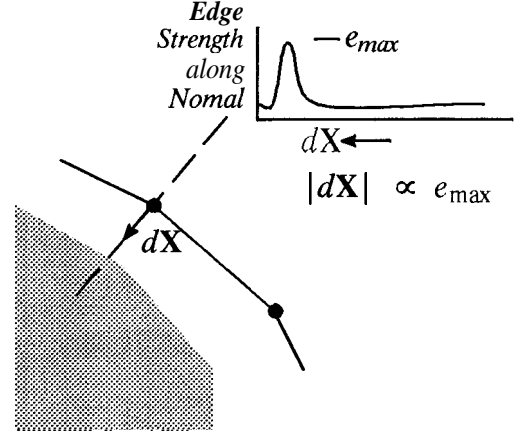

FIG. 21. Suggested movement of point is along normal to boundary, proportional to maximum edge strength on normal.

object we need to find a set of adjustments which will move each point toward a better position. When the model points represent the boundaries of objects (Fig. 20) this involves moving them toward the image edges. There are various approaches that could be taken. In the examples we describe below we use an adjustment along a normal to the model boundary toward the strongest image edge, with magnitude proportional to the strength of the edge (Fig. 21).

An alternative approach is to generate potential images such as those described by Kass *et al.* [3], possibly one for each model point, describing the likelihood of each point in the image being the model point. Adjustments to each point position can then be derived from the gradient of the potential image at the current estimate of the point's position.

However they are obtained, we denote the set of adjustments (Fig. 22) as a vector $d\mathbf{X}$, where
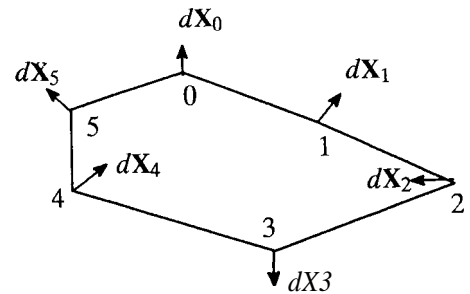
$$d\mathbf{X} = (dX_0, dY_0, \ldots, dX_{x-1}, dY_{n-1})^\mathrm{T}$$

FIG. 20. Part of a model boundary approximating to the edge of an image object.

FIG. 22. Adjustments to a set of points.

The vector **b** should lie within a hyperellipsoid about the origin. If updating **b** using (26) leads to an implausible shape, i.e., $D_m > D_{max}$ and the point lies outside the ellipsoid, **b** can be rescaled to lie on the closest point of the allowed volume using

$$b_k \rightarrow b_k \cdot \left(\frac{D_{max}}{D_m} \right) (k = 1, \ldots, t). \qquad (27)$$

Note that we have already applied implicit limits of zero to the weights of the eigenvectors truncated from our representation (i.e., $b_i = 0 \, \forall \, i > t$). Once the parameters have been updated, and limits applied where necessary, the updated positions of the model points can be calculated, and new suggested movements derived for each point. The procedure is repeated until no significant change results.

4.4. EXAMPLES USING ACTIVE SHAPE MODELS

The techniques described above have been used successfully in a number of applications, both industrial and medical [26, 27, 33]. Here we show results using the resistor, heart,, and hand models described above.

In each case initial estimates of the position, orientation, and scale are made, and the shape parameters of the Active Shape Model (ASM) are initialized at zero $(b, = 0 \, (i = 1, \ldots, t))$. Suggested movements for each model point are calculated by finding the strongest edge (of the correct polarity) along the normal to the boundary at the point (see 4.1 and Fig. 21). Adjustments to the parameters are calculated and applied, and the process is repeated.

4.4.1. *Locating Resistors.* We have constructed a Point Distribution Model of a resistor, representing its boundary using 32 points (Section 3.4.1). Figure 23 shows an image of part of a printed circuit board with the resistor boundary model superimposed as it iterates toward a component in the image. We interpolate an additional 32 points, one between each pair of model points around the boundary, and calculate adjustments to each point by finding the strongest edge along profiles 20 pixels long centred at each point. We use a shape model with 5 degrees of freedom. Each iteration of the ASM takes about 0.015 s on a Sun Sparc 10 Workstation.

The method is effective in maintaining the global shape constraints of the model and works well, given a sufficiently good starting approximation; we discuss methods of obtaining such initial hypotheses elsewhere [26, 27].

4.4.2. *Locating Keart Ventricles.* Figure 24a shows an example of an echocardiogram. The left ventricle is in the top right of the imaged region. Figure 24b shows the initial placement of an instance of the 96 point heart chamber model described above (Section 3.4.2). Figure 24c shows the ASM after 80 iterations. After 200 iterations (Fig. 24d) the model gives a good fit to the data. The shape model used has 12 degrees of freedom. The adjustments to each point are calculated using the strongest edge in a smoothed image along a profile 40 pixels long centered on the point. Each ASM iteration takes about 0.03 s on a Sun Sparc 10 workstation. In this example the model is able to infer the position of the parts of the boundary where there are missing data (for example, the top of the ventricle) by using the knowledge of the expected shape combined with information from the areas of the image where good evidence for the ventricle wall can be found. Without the prior knowledge of the shape given by the model it would not be possible to delineate the ventricle boundary accurately. Further medical applications of the method are described in [33].

4.4.3. *Locating Hands.* We have constructed a Point Distribution Model of a hand, representing the boundary using 72 points (Section 3.4.3). Figure 25 shows an image of one of the author's hands amid some clutter and occlusion, and an example of the model iterating towards it. We calculate adjustments to each point by finding the strongest edge on a profile 35 pixels long centred on the point. The shape model has 8 degrees of freedom, and each ASM iteration takes about 0.02 seconds on a Sun Sparc 10 Workstation. The result demonstrates that the method can deal with clutter and limited occlusion.

5. DISCUSSION

The examples given above illustrate the main features of our approach. Using a single method, specialized only by training with an appropriate set of examples, we have been able to locate automatically a range of structures in complex, noisy, and cluttered images. Other examples reported elsewhere include faces [36], handwritten characters [36], anatomical structures in magnetic resonance images of the brain and abdomen [33], vertebrae in radiographs [33], parts of the foot in pressure images [38] and all the parts in an automobile brake assembly [34]. We discuss below some of the issues which arise from this work, including areas where further development is required.

5.1. *Point Distribution Models*

5.1.1. *Choice of Model Points and Training Examples.* It is important that landmark points be placed on the training images as accurately as possible. If a point is not in the correct position on each shape, the model will be unable to correctly represent the position of that point—it will include terms describing the noise caused by errors in point location. It is equally important to arrange that all the examples used to train the model are
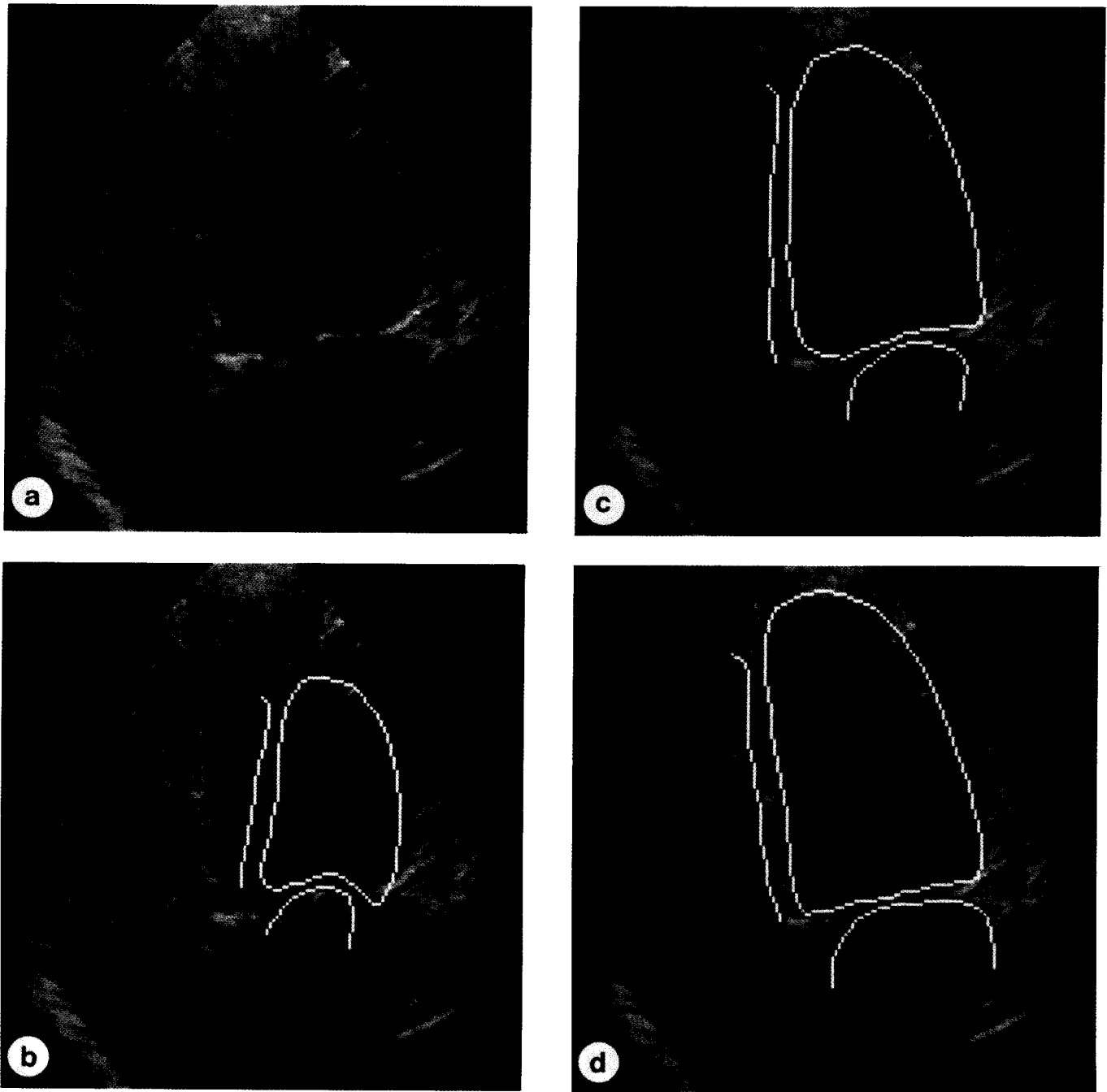
FIG. 24. Echocardiogram image with heart chamber boundary model superimposed, showing its initial position and its location after 80 and 200 iterations.

tools to ease the procedure. Techniques such as those described by Burr [29] and the Finite Element Models of Sclaroff and Pentland [30] or Nastar and Ayache [21] may be able to assist the user in locating point correspondences during this training phase.

In some cases occlusion and noise will lead to images in which some points cannot be accurately located. It is straightforward to adjust the calculation of mean shape (7) and the covariance matrix (9) to give a weighting to each point in each example in the training set. When some points are missing, the weights for known points can be set to unity; those for unknown points can be set to zero. As long as only a small proportion of points are missing in any one example, and no points are missing from all examples, it is still possible to build useful models.

In principle it is possible to "overtrain" a model. Sup-

pose that a large proportion of the examples were close to the mean and there were only one or two examples demonstrating some particular form of shape variation. It is possible that when the number of modes to be used, t, is chosen, the mode which best describes the infrequent shape variation will be truncated, since it will explain only a small amount of the total variance. However, since the training examples are typically selected and labeled by hand, it is time consuming and inefficient to include many similar shapes—it is better to choose a variety of different shapes which cover the whole range of variations one is likely to observe (where such are available). It is at this stage that the expert knowledge of a human can play a part.

5.1.2. Multipart Models. The heart example illustrates an important fact—that the points used to construct a PDM and its derived ASM do not need to belong to a single object or shape. The connectivity of the points is not relevant to the construction of the PDM and is only used by the ASM *to* determine the direction of the local normal at each point during image search. The shapes of multiple subparts of a complex assembly and the spatial relationships between them can thus be represented by a single PDM. A significant advantage of handling shape and spatial relationships in a unified way is that correlations between the positions and shapes of subcomponents can be modeled; this is important, for example, in assemblies of interlinked mechanical components or in medical images where several organs are "packed" into the same cavity.

5.1.3. Modeling Shape Variation. We showed in Section 3.3 that each aligned shape can be considered as a single point in $2n$ dimensional space, and the whole training set as a cloud of points in this space. We attempt to model this cloud using the idea of an Allowable Shape Domain. For the search method to work effectively it is important that this domain be simply connected, and that we have a simple method of navigating around the domain. The assumption that the domain is an ellipsoid (or a box with the same axes) allows us to do this. However, under certain circumstances this is an inappropriate model. When there is a large degree of bending or relative rotation in the training set, nonlinear relationships between landmarks can give the cloud in the $2n$ dimensional space a "banana" shape or worse. Under these circumstances, as was demonstrated in Section 3.4.4, the ellipsoidal assumption gives a shape model which can generate shapes badly distorted when compared with those from the training set. The model is not as *specific* as one would like, and only a subset of the shapes it can generate would be considered "legal." In some situations this is not disastrous. For instance, the worm model given can be used successfully to locate examples of worms in images, but

the models are more susceptible to being distorted by noise or clutter than a more specific model would be.

A more general model of the allowable shape domain could lead to more specific shape models. We have experimented using polynomials, rather than straight lines, for the axes of the domain with encouraging results. Instead of each mode defining straight line motion for each point, the points follow polynomial curves as the parameter varies. Results will be presented in a further paper.

5.1.4. *Dealing with Small Numbers of Examples.* If there are fewer training examples, N, than point coordinates ($2n$), as is often the case, particularly for complex models, there can be no more than $N - 1$ degrees of freedom in the model. The principal component analysis required for the method uses the eigenvectors of the $2n \times 2n$ matrix $\mathbf{S}$ (Eqs. 9, 10). When $N < 2n$ this matrix has no more than $N - 1$ nonzero eigenvalues. Calculating all $2n$ eigenvectors in this case is unnecessary. An efficient way of calculating the eigenvectors associated with nonzero eigenvalues is given in Appendix B.

5.1.5. Extensions to the Model. Rather than have one "flat" PDM representing a complex assembly, it is possible to build a hierarchical PDM in which the top layer controls the position, scale, orientation, and shape parameters of the layer below. The bottom layer can consist of a number of subcomponents, each represented by a "flat" PDM. Varying the parameters of the top layer varies the pose, scale, and shape of the various components below. This avoids problems with the PDM due to rotating subcomponents—their orientation relative to the rest of the assembly can be modeled explicitly, rather than implicitly in a single-layer linear PDM.

It is also easy to extend the Point Distribution Model to deal with three dimensional data, for example, 3D medical images. We have recently described a successful system for automated interpretation of 3D Magnetic Resonance images of the brain using a 3D PDM [35].

5.1.6. The Chord Length Distribution. Elsewhere we have described how to derive a shape model from a training set using the distances between pairs of points—a Chord Length Distribution Model [31]. The distance, R_{ij}, between every pair of points i, j in each example of the training set is calculated, and the way these chord lengths vary is modeled by calculating their mean and covariances and applying a Principal Component Analysis. A model with several parameters is obtained, which returns sets of interpoint distances, R_{ij}, from which a new shape can be constructed. Varying the parameters varies the distances, which causes the shape to change. Such a system is able to model the rigid parts of an object regardless of their orientation, since it relies only on internal distances. Though this technique is sometimes better than the linear PDM at representing objects which can bend (such as

on methods which incorporate the advantages of both approaches.

5.2.4. A Framework for Object Modeling and Recognition. We have conducted experiments which suggest that our local optimization method can be fruitfully used in conjunction with a Genetic Algorithm (GA) search [26–28]. The GA can be run as a cue generator to produce a number of object hypotheses, which can be refined using the Active Shape Model. Alternatively, the ASM can be combined with the GA search, applying one iteration at each generation of the Genetic Algorithm. Both techniques have been used successfully to locate complex structures in a variety of images.

6. CONCLUSIONS

We have described Point Distribution Models (PDMs)—statistical models of shape which can be constructed from training sets of correctly labeled images. A **PDM** represents an object as a set of labeled points, giving their mean positions and a small set of modes of variation which describe how the object's shape can change. Applying limits to the parameters of the model enforces global shape constraints ensuring that any new examples generated are similar to those in the training set. Given a set of shape parameters, an instance of the model can be calculated rapidly. The models are compact and are well suited to generate-and-test image search strategies.

Active Shape Models (ASMs) exploit the linear formulation of PDMs in an iterative search procedure capable of rapidly locating the modeled structures in noisy, cluttered images—even if they are partially occluded. Object identification and location are robust because the models are specific in the sense that instances are constrained to be similar to those in the training set.

We have demonstrated the ability to create compact models of resistors, hearts (in echocardiograms), and hands. We have also shown that these models can be used successfully in image search. Using a conventional workstation a good interpretation can typically be obtained in seconds. We have described elsewhere various other applications in which the same methods have been exploited successfully, including examples where very complex structures (e.g., faces and automobile brake assemblies) are modeled. The important point to stress is that precisely the same software can be applied to a broad range of image interpretation problems—both medical and industrial—specialized only by training with suitable examples.

We believe that this approach holds considerable promise as a practical but generic technique for automated image interpretation.

APPENDIX A: ALIGNING A PAIR OF SHAPES

Given two similar shapes, $\mathbf{x}_1$ and $\mathbf{x}_2$, we would like to choose a rotation, θ, a scale, s, and a translation, (t_x, t_y), mapping $\mathbf{x}_2$ onto $M(\mathbf{x}_2) + \mathbf{t}$ so as to minimize the weighted sum

$$E = (\mathbf{x}_1 - M(s, \theta)[\mathbf{x}_2] - \mathbf{t})^{\mathrm{T}}\mathbf{W}(\mathbf{x}_1 - M(s, \theta)[\mathbf{x}_2] - \mathbf{t}), \quad (3)$$

where

$$M(s, \theta)\begin{bmatrix} x_{jk} \\ y_{jk} \end{bmatrix} = \begin{pmatrix} (s\cos\theta)x_{jk} - (s\sin\theta)y_{jk} \\ (s\sin\theta)x_{jk} + (s\cos\theta)y_{jk} \end{pmatrix}, \quad (4)$$

$$\mathbf{t} = (t_x, t_y, \ldots, t_x, t_y)^{\mathrm{T}}, \quad (29)$$

and $\mathbf{W}$ is a diagonal matrix of weights for each point. If we write

$$a_x = s\cos\theta \quad a_y = s\sin\theta,$$

a least-squares approach (differentiating with respect to each of the variables a_x, a_y, t_x, t_y) leads to a set of four linear equations,

$$\begin{pmatrix} X_2 & -Y_2 & W & 0 \\ Y_2 & X_2 & 0 & W \\ Z & 0 & X_2 & Y_2 \\ 0 & Z & -Y_2 & X_2 \end{pmatrix}\begin{pmatrix} a_x \\ a_y \\ t_x \\ t_y \end{pmatrix} = \begin{pmatrix} X_1 \\ Y_1 \\ C_1 \\ C_2 \end{pmatrix}, \quad (30)$$

where

$$X_i = \sum_{k=0}^{n-1} w_k x_{ik} \quad Y_i = \sum_{k=0}^{n-1} w_k y_{ik} \quad (31)$$

$$Z = \sum_{k=0}^{n=1} w_k(x_{2k}^2 + y_{2k}^2) \quad W = \sum_{k=0}^{n-1} w_k \quad (32)$$

$$C_1 = \sum_{k=0}^{n-1} w_k(x_{1k}x_{2k} + y_{1k}y_{2k}) \quad (33)$$

$$C_2 = \sum_{k=0}^{n-1} w_k(y_{1k}x_{2k} - x_{1k}y_{2k}). \quad (34)$$

These can be solved for a_x, a_y, t_x, and t_y using standard matrix methods.

APPENDIX B: CALCULATING THE EIGENVECTORS OF THE COVARIANCE MATRIX WHEN THERE ARE FEWER SAMPLES THAN CO-ORDINATES

When there are fewer training examples, N, than point co-ordinates, $2n$, the eigenvectors of the $2n \times 2n$ covariance matrix S can be calculated from the eigenvectors of

19. A. Pentland and S. 'Sclaroff, Closed-form solutions for physically based modeling and recognition, *IEEE Trans. Pattern Anal. Mach. Intell. W*, 1991, 715–729.

20. D. Terzopoulos and D. Metaxas, Dynamic 3D models with local and global deformations: Deformable superquadrics, *IEEE Trans. Pattern Anal. Mach. Intell. W*, 1991, 703–714.

21. C. Nastar and N. Ayache, Fast segmentation, tracking and analysis of deformable objects, in *Proceedings, International Conference on Computer Vision, 1993*, pp. 275–279, IEEE Comput. Soc. Press, 1993.

22. P. Karaolani, G. D. Sullivan, K. D. Baker, and M. J. Baines, A finite element method for deformable models, in *Proceedings of the Fifth Alvey Vision Conference, Reading, 1989*, pp. 73–78.

23. P. Karaolani, G. D. Sullivan, and K. D. Baker, Active contours using finite elements to control local scale, in *Proceedings, British Machine Vision Conference 1992*, pp. 481–487, Springer-Verlag, Berlin/New York, 1992.

24. J. C. Gower, Generalized Procrustes analysis, *Psychometrika* **40**, 1975, 33–51.

25. R. A. Johnson and D. W. Wichern, *Multivariate Statistics, A Practical Approach*, Chapman & Hall, London/New York, 1988.

26. A. Hill, T. F. Cootes, and C. J. Taylor, A genetic system for image interpretation using flexible templates, in *British Machine Vision Conference*, Springer-Verlag, 1992.

27. A. Hill, C. J. Taylor, and T. Cootes, Object recognition by flexible template matching using genetic algorithms, in *Proceedings, European Conference on Computer Vision* (G. Sandini, Ed.), pp. 852–856, Springer-Verlag, Berlin/New York, 1992.

28. A. Hill, A. Thornham, and C. J. Taylor, Model-based interpretation of 3D medical images, in *Proceedings, British Machine Vision Conference 1993* (J. Illingworth, Ed.), Vol. 2, pp. 339–348, BMVA Press, 1993.

29. D. J. Burr, A dynamic model for image registration, *Comput. Graphics Image Process.* **15**, 1981, pp. 102–112.

30. S. Sclaroff and A. Pentland, A model framework for correspondence and description, in *Proceedings, International Conference on Computer Vision, 1993*, pp. 715–729, IEEE Comput. Soc. Press, 1993.

31. T. F. Cootes, D. H. Cooper, C. J. Taylor, and J. Graham, A trainable method of parametric shape description, *Image Vision Comput.* **10**, 1992, 289–294.

32. T. F. Cootes and C. J. Taylor, Active shape model search using local grey-level models: A quantitative evaluation, in *Proceedings, British Machine Vision Conference 1993* (J. Illingworth, Ed.), Vol. 2, pp. 639–648, BMVA Press, 1993.

33. T. F. Cootes, A. Hill, C. J. Taylor, and J. Haslam, The use of active shape models for locating structures in medical images, in *Proceedings, Information Processing for Medical Imaging* (H. H. Barrett and A. F. Gmitro, Ed.), pp. 33–47, Springer-Verlag, Berlin/New York, 1993.

34. T. F. Cootes, C. J. Taylor, A. Lanitis, D. H. Cooper, and J. Graham, Building and using flexible models incorporating grey-level information, in *Proceedings, International Conference on Computer Vision*, pp. 242–246, IEEE Comput. Soc. Press, 1993.

35. A. Hill, A. Thornham, and C. J. Taylor, Model-based interpretation of 3D medical images, in *Proceedings, British Machine Vision Conference, 1993* (J. Illingworth, Ed.), Vol. 1, pp. 339–348, BMVA Press, 1993.

36. A. Lanitis, C. J. Taylor, and T. F. Cootes, A generic system for classifying variable objects using flexible template matching, in *Proceedings, British Machine Vision Conference, 1993* (J. Illingworth, Ed.), Vol. 1, pp. 329–338, BMVA Press, 1993.

37. D. G. Lowe, Fitting parameterized three-dimensional models to images, *IEEE Trans. Pattern Anal. Mach. Intell.* **13**, 1991, 441–450.

38. J. A. Grogan, *Automated Analysis of Pedobarograph Images*, M. Sc. Thesis, Victoria University of Manchester, Oct. 1993.