Constrained Active Appearance Models

T. F. Cootes and C. J. Taylor
Dept. Imaging Science and Biomedical Engineering,
University of Manchester,
Manchester M13 9PT U.K.
t.cootes@man.ac.uk

Abstract

Active Appearance Models (AAMs) have been shown to be useful for interpreting images of deformable objects. Here we place the AAM matching algorithm in a statistical framework, allowing extra constraints to be applied. This enables the models to be combined with other methods of object location. We demonstrate how user interaction can be used to guide the search and give results of experiments showing the effect of constraints on the performance of model matching.

1. Introduction

Statistical models of appearance have been shown to be useful for interpreting images of deformable objects, particularly when combined with a fast matching algorithm such as Active Appearance Model (AAM) search [4]. In many practical applications an AAM alone is insufficient. A suitable initialisation is required for the matching process and when unconstrained the AAM may not always converge to the correct solution. The appearance model provides shape and texture information which are combined to generate a model instance. A natural approach to initialisation and constraint is to provide prior estimates of the position of some of the shape points, either manually or using automatic feature detectors. For instance, when matching a face model it is useful to have an estimate of the positions of the eyes, which could either be provided by a user or located using a suitable eye detector.

This paper reformulates the original least squares matching of the AAM search algorithm into a statistical framework. This allows the introduction of prior probabilities on the model parameters and the inclusion of prior constraints on point positions. The latter allows one or more points to be pinned down to particular positions with a given variance. This framework enables the AAM to be integrated with other feature location tools in a principled manner, as long as those tools can provide an estimate of the error on their output.

In the following we describe the mathematics in detail, give examples of using point constraints to help user guided image markup and give the results of quantitative experiments studying the effects of constraints on image matching.

1.1. Previous Work

Statistical models of appearance which combine shape and texture variation have been described by various groups including Edwards *et. al.* [5] and Jones and Poggio [9]. Edwards *et. al.* initially matched the models to new images using an 'Active Shape Model' approach [3], but later developed the Active Appearance Model [4]. Many groups have placed deformable model matching in a statistical framework, for instance [7, 1, 2]. Hug *et. al.* [8] describe how constraints can be applied to statistical models of shape. They show how to generate the reduced set of modes of variation obtained when a subset of the model points are constrained, and find the optimal sets of points to further restrict shape variation. Fua and Miccio [6] matched 3D mesh head models to 2D images, and incorporated point constraints

Mitchell *et. al.* [11] used a combination of Active Shape Model and Active Appearance model to segment cardiac images. At each iteration the two models ran independently to compute new estimates of the pose and shape parameters. These were then combined using a weighted average. They showed that this approach gave better results than the AAM alone. The approach described below gives a more principled method of combining the algorithms.

1.2. Statistical Appearance Models

Here we describe briefly the models of shape and appearance used to represent classes of deformable objects, so as to establish mathematical notation. Statistical appearance models are generated by combining a model of shape variation with a model of the texture variation - in the following 'texture' means the pattern of intensities or colours across an image patch. To build a model a training set of annotated images is required, in which corresponding points have been marked on each example. The points are each represented as a vector, $\mathbf{x}$, aligned into a common co-ordinate frame and a statistical shape model is built [3]. Each training image is then warped so that the points match those of the mean shape, obtaining a 'shape-free patch'. This is raster scanned into a texture vector, $\mathbf{g}$, which is then normalized. Eigen-analysis is applied to build a texture model. Finally the correlations between shape and texture are learnt to generate a combined appearance model.

The appearance model has parameters, $\mathbf{c}$, controlling the shape and texture (in the model frame) according to

$$\begin{array}{rcl} \mathbf{x} & = & \bar{\mathbf{x}} + \mathbf{Q}_s\mathbf{c} \\ \mathbf{g} & = & \bar{\mathbf{g}} + \mathbf{Q}_g\mathbf{c} \end{array} \quad (1)$$

where $\bar{\mathbf{x}}$ is the mean shape, $\bar{\mathbf{g}}$ the mean texture in a mean shaped patch and $\mathbf{Q}_s$,$\mathbf{Q}_g$ are matrices describing the modes of variation derived from the training set.

A shape in the image frame, $\mathbf{X}$, can be generated by applying a suitable transformation to the points, $\mathbf{x} : \mathbf{X} = S_{\mathbf{t}}(\mathbf{x})$. Typically $S_{\mathbf{t}}$ will be a similarity transformation described by a scaling, s, an in-plane rotation, θ, and a translation (t_x, t_y). For linearity the scaling and rotation can be represented as (s_x, s_y) where $s_x = (s\cos\theta - 1)$, $s_y = s\sin\theta$. The pose parameter vector $\mathbf{t} = (s_x, s_y, t_x, t_y)^T$ is then zero for the identity transformation.

The texture in the image frame is generated by applying a scaling and offset to the intensities, $\mathbf{g}_{im} = T_{\mathbf{u}}(\mathbf{g}) = (u_1 + 1)\mathbf{g}_{im} + u_2\mathbf{1}$, where $\mathbf{u}$ is the vector of transformation parameters.

A full reconstruction is given by generating the texture in a mean shaped patch, then warping it so that the model points lie on the image points, $\mathbf{X}$.

2. Model Matching

Model matching can be treated as an optimisation process, minimising the difference between the synthesized model image and the target image. The appearance model parameters, $\mathbf{c}$, and shape transformation parameters, $\mathbf{t}$, define the position of the model points in the image frame, $\mathbf{X}$, which gives the shape of the image patch to be represented by the model. To test the quality of the match with the current parameters, the pixels in the region of the image defined by $\mathbf{X}$ are sampled to obtain a texture vector, $\mathbf{g}_{im}$. These are projected into the texture model frame, $\mathbf{g}_s = T^{-1}(\mathbf{g}_{im})$. The current model texture is given by $\mathbf{g}_m = \bar{\mathbf{g}} + \mathbf{Q}_g\mathbf{c}$. The current difference between model and image (measured in the normalized texture frame) is thus

$$\mathbf{r}(\mathbf{p}) = \mathbf{g}_s - \mathbf{g}_m \quad (2)$$

where $\mathbf{p}$ is a vector containing the parameters of the model, $\mathbf{p}^T = (\mathbf{c}^T|\mathbf{t}^T|\mathbf{u}^T)$.

2.1. Basic AAM Formulation

The original formulation of Active Appearance Models aimed to minimise a sum of squares of residuals measure, $E_1(\mathbf{p}) = \mathbf{r}^T\mathbf{r}$. Suppose that during an iterative matching process the current residual was $\mathbf{r}$. We wish to choose an update $\delta\mathbf{p}$ so as to minimize $E_1(\mathbf{p} + \delta\mathbf{p})$. By applying a first order Taylor expansion to (2) we can show that $\delta\mathbf{p}$ must satisfy

$$\frac{\partial\mathbf{r}}{\partial\mathbf{p}}\delta\mathbf{p} = -\mathbf{r} \quad (3)$$

where the ij^{th} element of matrix $\frac{\partial\mathbf{r}}{\partial\mathbf{p}}$ is $\frac{dr_i}{dp_j}$. Thus

$$\delta\mathbf{p} = -\mathbf{R}\mathbf{r}(\mathbf{p}) \quad (4)$$

where

$$\mathbf{R}_1 = (\frac{\partial\mathbf{r}}{\partial\mathbf{p}}^T\frac{\partial\mathbf{r}}{\partial\mathbf{p}})^{-1}\frac{\partial\mathbf{r}}{\partial\mathbf{p}}^T \quad (5)$$

In a standard optimization scheme it would be necessary to recalculate $\frac{\partial\mathbf{r}}{\partial\mathbf{p}}$ at every step, an expensive operation. However, it is assumed that since it is being computed in a normalized reference frame, it can be considered approximately fixed. It is thus estimated once from the training set. $\frac{\partial\mathbf{r}}{\partial\mathbf{p}}$ can be estimated by numeric differentiation, systematically displacing each parameter from the known optimal value on typical images and computing an average over the training set. Residuals at displacements of differing magnitudes are measured (typically up to 0.5 standard deviations of each parameter) and combined with a Gaussian kernel to smooth them. $\mathbf{R}$ and $\frac{\partial\mathbf{r}}{\partial\mathbf{p}}$ are precomputed and used in all subsequent searches with the model.

2.2. MAP Formulation

Rather than simply minimising a sum of squares measure, we can put the model matching in a probabalistic framework. In a maximum a-posteriori (MAP) formulation we seek to maximise

$$p(model|data) \propto p(data|model)p(model) \quad (6)$$

If we assume that the residual errors can be treated as uniform gaussian with variance σ_r^2, and that the model parameters are gaussian with diagonal covariance $\mathbf{S}_p^2$, then maximising (6) is equivalent to minimising

$$E_2(\mathbf{p}) = \sigma_r^{-2}\mathbf{r}^T\mathbf{r} + \mathbf{p}^T(\mathbf{S}_p^{-1})\mathbf{p} \quad (7)$$

If we form the combined vector

$$\mathbf{y} = \begin{pmatrix} \sigma_r^{-1}\mathbf{r} \\ \mathbf{S}_p^{-0.5}\mathbf{p} \end{pmatrix} \tag{8}$$

then $E_2 = \mathbf{y}^T\mathbf{y}$ and

$$\mathbf{y}(\mathbf{p}+\delta\mathbf{p}) = \mathbf{y}(\mathbf{p}) + \begin{pmatrix} \sigma_r^{-1}\frac{\partial \mathbf{r}}{\partial \mathbf{p}} \\ \mathbf{S}_p^{-0.5} \end{pmatrix} \delta\mathbf{p} \tag{9}$$

In this case the optimal update step is the solution to the equation

$$\begin{pmatrix} \sigma_r^{-1}\frac{\partial \mathbf{r}}{\partial \mathbf{p}} \\ \mathbf{S}_p^{-1} \end{pmatrix} \delta\mathbf{p} = - \begin{pmatrix} \sigma_r^{-1}\mathbf{r}(\mathbf{p}) \\ \mathbf{S}_p^{-0.5}\mathbf{p} \end{pmatrix} \tag{10}$$

which can be shown to have the form

$$\delta\mathbf{p} = -(\mathbf{R}_2\mathbf{r} + \mathbf{K}_2\mathbf{p}) \tag{11}$$

where $\mathbf{R}_2$ and $\mathbf{K}_2$ can be precomputed during training. In this case the update step consists just of image sampling and matrix multiplication.

2.3. Including Priors on Point Positions

Suppose we have prior estimates of the positions of some points in the image frame, $\mathbf{X}_0$, together with their with covariances $\mathbf{S}_X$. Unknown points will give zero values in appropriate positions in $\mathbf{X}_0$ and effectively infinite variance, leading to suitable zero values in the inverse covariance, $\mathbf{S}_X^{-1}$. Let $\mathbf{d}(\mathbf{p}) = (\mathbf{X} - \mathbf{X_0})$ be a vector of the displacements of the current point positions from those target points. A measure of quality of fit (related to the log-probability) is then

$$E_3(\mathbf{p}) = \sigma_r^{-2}\mathbf{r}^T\mathbf{r} + \mathbf{p}^T(\mathbf{S}_p^{-1})\mathbf{p} + \mathbf{d}^T\mathbf{S}_X^{-1}\mathbf{d} \tag{12}$$

Following a similar approach to that above we obtain the update step as a solution to

$$\mathbf{A}_1\delta\mathbf{p} = -\mathbf{a}_1 \tag{13}$$

where

$$\begin{aligned} \mathbf{A}_1 &= \left(\sigma_r^{-2}\frac{\partial \mathbf{r}}{\partial \mathbf{p}}^T\frac{\partial \mathbf{r}}{\partial \mathbf{p}} + \mathbf{S}_p^{-1} + \frac{\partial \mathbf{d}}{\partial \mathbf{p}}^T\mathbf{S}_X^{-1}\frac{\partial \mathbf{d}}{\partial \mathbf{p}}\right) \\ \mathbf{a}_1 &= \left(\sigma_r^{-2}\frac{\partial \mathbf{r}}{\partial \mathbf{p}}^T\mathbf{r}(\mathbf{p}) + \mathbf{S}_p^{-1}\mathbf{p} + \frac{\partial \mathbf{d}}{\partial \mathbf{p}}^T\mathbf{S}_X^{-1}\mathbf{d}\right) \end{aligned} \tag{14}$$

When computing $\frac{\partial \mathbf{d}}{\partial \mathbf{p}}$ one must take into account the global transformation as well as the shape changes, the parameters of which have been folded into the vector $\mathbf{p}$ in the above to simplify the notation. The positions of the model points in the image plane are given by

$$\mathbf{X} = S_{\mathbf{t}}(\bar{\mathbf{x}} + \mathbf{Q}_s\mathbf{c}) \tag{15}$$

Therefore $\frac{\partial \mathbf{d}}{\partial \mathbf{p}} = (\frac{\partial \mathbf{d}}{\partial \mathbf{c}} | \frac{\partial \mathbf{d}}{\partial \mathbf{t}})$, where

$$\frac{\partial \mathbf{d}}{\partial \mathbf{c}} = \frac{\partial S_{\mathbf{t}}(\mathbf{x})}{\partial \mathbf{x}}\mathbf{Q}_s \quad , \quad \frac{\partial \mathbf{d}}{\partial \mathbf{t}} = \frac{\partial S_{\mathbf{t}}(\mathbf{x})}{\partial \mathbf{t}} \tag{16}$$

These can be substituted into (13) and (14) to obtain the update steps. Note that in this case a linear system of equations must be solved at each iteration. See Appendix A for the special case of isotropic errors on the fixed point position estimates.

3. Experiments

To test the effects of applying constraints to the AAM search, we performed a series of experiments on images from the M2VTS face database [10]. We trained an AAM on 100 images and tested it on another 100 images. 68 points were used to define the face shape, and 5000 grey-scale pixels used to represent the texture across the face patch. The separation between the eyes is about 100 pixels (see Figure 10). The model used 56 appearance parameters.

3.1. User Guided Search

Figure 10a shows the initial point positions and reconstruction of a face model on a new image. Figure 10b shows the result after an unconstrained search. In this case the model fails because the eyebrows match to the eyes. Figure 10c shows the result after the user constrains the right eye centre. Figure 10d shows the result after the user constrains a point on the left eyebrow. This gives a much better overall match. If necessary further points can be pinned, depending on the overall accuracy required. Each search takes about 500ms (on a Pentium 550MHz PC), allowing efficient user interaction.

3.2. Point Constraints

To test the effect of constraining points, we assumed that the position of the eye centres was known to a given variance. We then displaced the model from the correct position by up to 10 pixels (5% of face width) in x and y and ran a constrained search. Nine searches were performed on each of the 100 unseen test images. Measurements were made of the boundary error (the RMS separation in pixels between the model points and hand-marked boundaries) and the RMS texture error in grey level units (the images have grey-values in the range [0,255]). The eye centres were assumed known to various accuracies, defined in terms of a variance on their position.

Figure 1 shows the effect on the boundary error of varying the variance estimate. Figure 2 shows the effect on texture error. The results suggest that there is an optimal value of error standard deviation of about 7 pixels. Pinning down the eye points too harshly reduces the ability of the model to match accurately elsewhere.

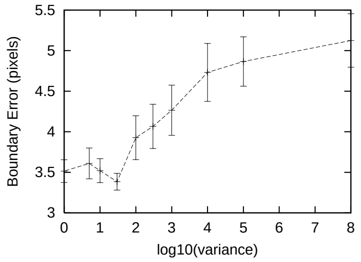


Figure 1. Effect on boundary error of varying variance on two eye centres

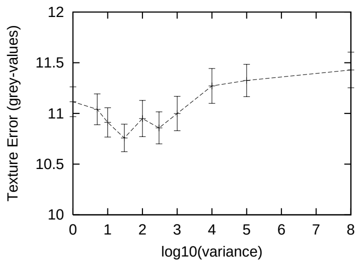


Figure 2. Effect on texture error of varying variance on two eye centres

To test the effect of errors on the positions of the fixed points, we repeated the above experiment, but randomly perturbed the fixed points with isotropic gaussian noise. This simulates inaccuracies in the output of feature detectors that might be used to find the points. Figure 3 shows the boundary error for different noise variances. The known noise variance was used as the constraining variance on the points. This shows that for small errors on the fixed points the search gives better results. However as the positional errors increase the fixed points are effectively ignored (a large variance gives a small weight), and the boundary error gets no worse.

3.3. Including Priors on the Parameters

We repeated the above experiment, but added a term giving a gaussian prior on the model parameters, $\mathbf{c}$ - equation (12). Figures 4, 5 show the results. There is an improvement in the positional error, but the texture error becomes worse.

Figure 6 shows the distribution of boundary errors at search convergence when matching is performed with/without priors on the parameters and with/without fixing the eye points. Figure 7 shows the equivalent texture

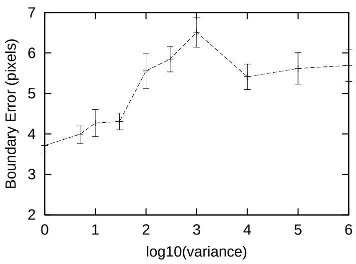


Figure 3. Boundary error vs eye centre variance

errors. Again, the most noticable effect is that including a prior on the parameters significantly increases the texture error, though there is little change to the boundary error. Adding the prior biases the match toward the mean. Small changes in point positions (eg away from edges) can introduce large changes in texture errors, so the image evidence tends to resist point movement towards the mean. However, there is less of a gradient for small changes in parameters which affect texture, so they tend to move toward the mean more readily.

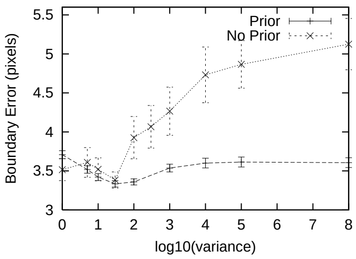


Figure 4. Boundary error vs varying eye centre variance, using a prior on the model parameters

3.4. Varying Number of Points

The experiment was repeated once more, this time varying the number of point constraints. Figure 8 shows the boundary error as a function of the number constrained points. The eye centres were fixed first, then the mouth, the chin and the sides of the face. There is a gradual improvement as more points are added. Figure 9 shows the texture error. This improves at first, as the model is less likely to fail to converge to the correct result with more constraints. However, when more points are added the texture error begins to increase slightly, as the texture match becomes compromised in order to satisfy the point constraints.

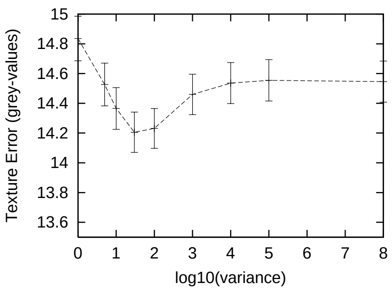


Figure 5. Texture error vs varying eye centre variance, using a prior on the model parameters

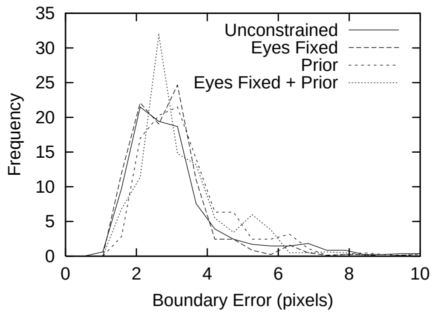


Figure 6. Distribution of boundary errors given constraints on points and parameters

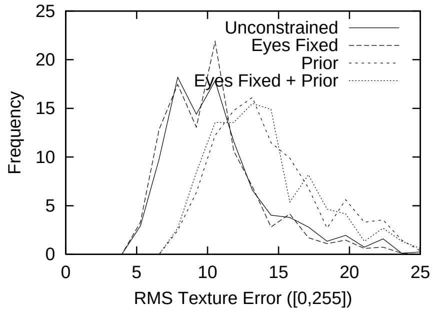


Figure 7. Distribution of texture errors given constraints on points and parameters

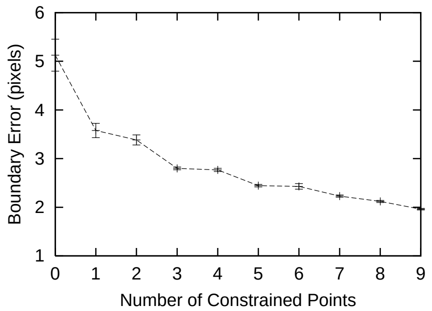


Figure 8. Boundary error vs Number of points constrained

4. Discussion and Conclusions

We have reformulated the AAM matching algorithm in a statistical framework, allowing it to be combined with user input and other tools in a principled manner. This is very useful for real applications where several different algorithms maybe required to obtain robust initialisation and matching.

Providing constraints on some points can improve the reliability and accuracy of the model matching. Adding a prior to the model parameters can improve the mean boundary position error, but at the expense of a significant degradation of the overall texture match.

The ability to pin down points interactively is useful for interactive model matching. The appearance models require labelled training sets, which are usually generated manually. A 'bootstrap' approach can be used in which the current model is used to help mark up new images, which are then added to the model. The interactive search makes this process significantly quicker and easier.

We anticipate that this framework will allow the AAMs to be used more effectively in practical applications.

A. Isotropic Point Errors

We consider the special case in which the fixed points are assumed to have positional variance equal in each direction, thus the covariance, $\mathbf{S}_X$ is diagonal. Suppose the shape transformation $S_{\mathbf{t}}(\mathbf{x})$ is a similarity transformation which scales by s.

Let $\mathbf{x}_0 = S_{\mathbf{t}}^{-1}(\mathbf{X}_0)$ and $\mathbf{y} = s(\mathbf{x} - \mathbf{x}_0)$. Then $\mathbf{d}^T\mathbf{S}_X^{-1}\mathbf{d} = \mathbf{y}^T\mathbf{S}_X^{-1}\mathbf{y}$.

If assume we all parameters are equally likely, to simplify notation, then (12) becomes

$$E_3(\mathbf{p}) = \sigma_r^{-2}\mathbf{r}^T\mathbf{r} + \mathbf{y}^T\mathbf{S}_X^{-1}\mathbf{y} \tag{17}$$

The update step is the solution to

$$\mathbf{A}_3\delta\mathbf{p} = -\mathbf{a}_3 \tag{18}$$

where

$$\begin{array}{rcl} \mathbf{A}_3 & = & \left(\sigma_r^{-2}\frac{\partial\mathbf{r}}{\partial\mathbf{p}}^T\frac{\partial\mathbf{r}}{\partial\mathbf{p}} + \frac{\partial\mathbf{y}}{\partial\mathbf{p}}^T\mathbf{S}_X^{-1}\frac{\partial\mathbf{y}}{\partial\mathbf{p}}\right) \\ \mathbf{a}_3 & = & \left(\sigma_r^{-2}\frac{\partial\mathbf{r}}{\partial\mathbf{p}}^T\mathbf{r}(\mathbf{p}) + \frac{\partial\mathbf{y}}{\partial\mathbf{p}}^T\mathbf{S}_X^{-1}\mathbf{y}\right) \end{array} \tag{19}$$

and $\frac{\partial\mathbf{y}}{\partial\mathbf{p}} = (\frac{\partial\mathbf{y}}{\partial\mathbf{c}}|\frac{\partial\mathbf{y}}{\partial\mathbf{t}})$, $\frac{\partial\mathbf{y}}{\partial\mathbf{c}} = s\mathbf{Q}_s$.

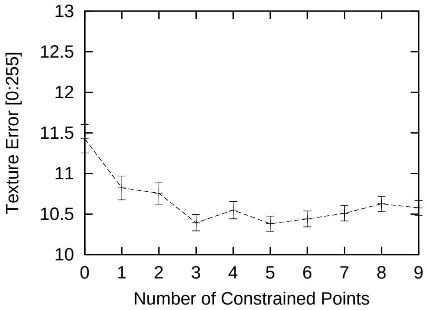


Figure 9. Texture errors vs Number of points constrained

$$\frac{\partial \mathbf{y}}{\partial \mathbf{t}} = -s\frac{\partial(S_{\mathbf{t}}^{-1}(\mathbf{X}_0))}{\partial \mathbf{t}} - (\mathbf{x}-\mathbf{x}_0).(\frac{s_x}{s}, \frac{s_y}{s}, 0, 0) \quad (20)$$

References

[1] A. Blake and M. Isard. *Active Contours*. Springer, 1998.

[2] C. Chesnaud, P. Refregier, and V. Boulet. Statistical region snake-based segmentation adapted to different physical noise models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(11):1145–57, 1999.

[3] T. F. Cootes, C. J. Taylor, D. Cooper, and J. Graham. Active shape models - their training and application. *Computer Vision and Image Understanding*, 61(1):38–59, Jan. 1995.

[4] G. Edwards, C. J. Taylor, and T. F. Cootes. Interpreting face images using active appearance models. In 3^{rd} *International Conference on Automatic Face and Gesture Recognition 1998*, pages 300–305, Japan, 1998.

[5] G. J. Edwards, C. J. Taylor, and T. F. Cootes. Learning to identify and track faces in image sequences. In 8^{th} *British Machine Vison Conference*, pages 130–139, Colchester, UK, 1997.

[6] P. Fua and C. Miccio. From regular images to animated heads: A least squares approach. In H.Burkhardt and B. Neumann, editors, 5^{th} *European Conference on Computer Vision*, volume 1, pages 188–202. Springer, Berlin, 1998.

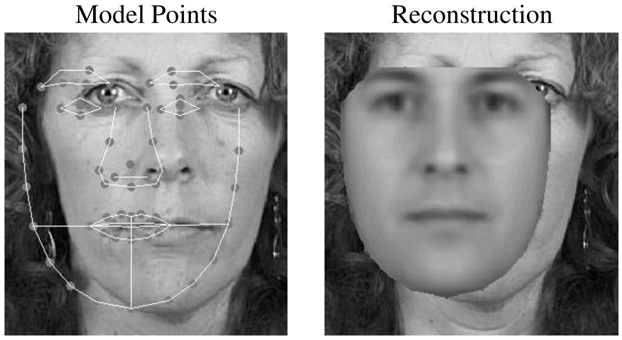
[7] U. Grenander and M. Miller. Representations of knowledge in complex systems. *Journal of the Royal Statistical Society B*, 56:249–603, 1993.

[8] J. Hug, C. Brechbuhler, and G. Szekely. Model-based initialisation for segmentation. In 6^{th} *European Conference on Computer Vision*, volume 2, pages 290–306. Springer, 2000.

[9] M. J. Jones and T. Poggio. Multidimensional morphable models. In 6^{th} *International Conference on Computer Vision*, pages 683–688, 1998.

[10] K. Messer, J. Matas, J. Kittler, J. Luettin, and G. Maitre. Xm2vtsdb: The extended m2vts database. In *Proc. 2nd Conf. on Audio and Video-based Biometric Personal Verification*. Springer Verlag, 1999.

[11] S. Mitchell, B.Lelieveldt, R. van der Geest, J. Schaap, J. Reiber, and M. Sonka. Segmentation of cardiac mr images: An active appearance model approach. In *SPIE Medical Imaging*, Feb. 2000.



a) Initial position for model on new image

b) Result of unconstrained AAM search

c) Right eye centre constrained

d) Right eye centre and left eyebrow point fixed

Figure 10. Example of user guided search. Initial unconstrained search is unsatisfactory. Pinning down some points leads to improved results.