Interpreting Face Images using Active Appearance Models

G.J. Edwards, C.J. Taylor and T.F. Cootes
Wolfson Image Analysis Unit,
Department of Medical Biophysics,
University of Manchester,
Manchester M13 9PT, U.K.
gje@sv1.smb.man.ac.uk

Abstract

We demonstrate a fast, robust method of interpreting face images using an Active Appearance Model (AAM). An AAM contains a statistical model of shape and grey-level appearance which can generalise to almost any face. Matching to an image involves finding model parameters which minimise the difference between the image and a synthesised face. We observe that displacing each model parameter from the correct value induces a particular pattern in the residuals. In a training phase, the AAM learns a linear model of the correlation between parameter displacements and the induced residuals. During search it measures the residuals and uses this model to correct the current parameters, leading to a better fit. A good overall match is obtained in a few iterations, even from poor starting estimates. We describe the technique in detail and show it matching to new face images.

1 Introduction

There is currently a great deal of interest in model-based approaches to the interpretation of face images [9] [4] [7] [6][3]. The attractions are two-fold: robust interpretation is achieved by constraining solutions to be face-like; and the ability to 'explain' an image in terms of a set of model parameters provides a natural interface to applications of face recognition. In order to achieve these objectives, the face model should be as complete as possible - able to synthesise a very close approximation to any face image which will need to be interpreted.

Although model-based methods have proved quite successful, none of the existing methods uses a full, photo-realistic model and attempts to match it directly by minimising the difference between the model-synthesised face and the image under interpretation. Although suitable photo-realistic models exist, (e.g. Edwards *et al* [3]), they typically involve a very large number of parameters (50-100) in order to deal with the variability due to differences between individuals, and changes in pose, expression, and lighting. Direct optimisation over such a high dimensional space seems daunting.

In this paper, we show that a direct optimisation approach is feasible and leads to an algorithm which is rapid, accurate, and robust. In our proposed method, we do not attempt to solve a general optimisation each time we wish to fit the model to a new face image. Instead, we exploit the fact the optimisation problem is similar each time - we can learn these similarities off-line. This allows us to find rapid directions of convergence even though the search space has very high dimensionality. In this paper we discuss the idea of image interpretation by synthesis and describe previous related work. In section 2 we explain how we build compact models of face appearance which are capable of generating synthetic examples of any individual, showing any expression, under a range of poses, and under any lighting conditions. We then describe how we rapidly generate face hypotheses giving possible locations and approximate scales. In section 4 we describe our Active Appearance Model algorithm in detail and in 5 demonstrate its performance.

1.1 Interpretation by Synthesis

In recent years many model-based approaches to the interpretation of face images have been described. One motivation is to achieve robust performance by using the model to constrain solutions to be face-like. A model also provides the basis for a broad range of applications by 'explaining' the appearance of a given face image in terms of a compact set of model parameters. These parameters are often used to characterize the identity, pose or expression of a face. In order to interpret a new image, an efficient method of finding the best match between image and model is required.

Models which can synthesise full faces have been described by several authors. Turk and Pentland [9] developed the 'eigenface' approach. However, this is not robust to shape changes in faces, and does not deal well with variability in pose and expression. Ezzat and Poggio [4] synthesise new views of a face from a set of example views, but cannot generalize to unseen faces. Nastar *et al* [7] use a 3D model of the grey-level surface, allowing full synthesis of shape and appearance. However the proposed search algorithm is likely to get stuck in local minima so is not robust. Lanitis *et al* [6] used separate models of shape and the local grey-level appearance of a 'shape-normalised' face. Edwards *et al* [3] extended this by also modelling the correlations between shape and grey-level appearance. Fitting such models to new images is achieved in most cases by minimising an error measure between the predicted appearance and the image, and is typically time consuming when the full model is used. Edwards *et al*[3] follow Lanitis *et al* [6] in using an Active Shape Model to find the face shape quickly. They then warp the image into a normalised frame and fit a model of the grey-level appearance to the whole face in this frame. This is effective, but as the ASM search does not use all the information available, it is not always robust. Our new approach can be seen as an extension of this idea, using all the information in a full appearance model to fit to the image. Our aim is take appearance models similar those described by Edwards *et al* [3] and fit them directly to face images. These models are both specific and detailed, allowing a complete description of a new face. By using all the information available, we expect to obtain robust performance. This approach involves a very high dimensional search problem, but we show below that an efficient method of solution exists. Efficient stochastic methods of fitting rigid models to images have been described by Viola and Wells [10] and Matas *et al* [5]. We adopt a similar strategy for generating face hypotheses when we have no initial knowledge of where the face may lie in an image. Given a hypothesis, we must refine it to obtain a better fit to the image. This involves estimating both the shape and the grey-level appearance of the face. Covell [2] demonstrated that the parameters of an eigen-feature model can be used to drive shape model points to the correct place. Similarly, Black and Yacoob [1] used local, hand-crafted models of image flow to track facial features. We use a generalisation of these ideas, using a model which relates the match residual to the error in the appearance parameters.

In a parallel development Sclaroff and Isidoro [8], have demonstrated 'Active Blobs' for tracking. The approach is broadly similar in that they use image differences to drive tracking, learning the relationship between image error and parameter offset in an off-line processing stage. The main difference is that Active Blobs are derived from a single example, whereas Active Appearance Models use a training set of examples. Sclaroff and Isidoro are primarily interested in tracking and use an initial frame as a template. They assume that the object being tracked may be non-rigid, or that projective effects may render it so in the image plane, and allow deformations consistent with low energy mesh distortion (derived using a Finite Element method). A simple polynomial model is used to allow changes in intensity across the object. Active Appearance Models learn what are valid shape and intensity variations from their training set.

Sclaroff and Isidoro suggest applying a robust kernel to the image differences, an idea we will use in later work. Also, since annotating the training set is the most time consuming part of building an AAM, the Active Blob approach may be useful for 'bootstrapping' from the first example.

2 Modelling Facial Appearance

In this section we outline how our facial appearance models were generated. The approach follows that described in Edwards *et al* [3] to which the reader is directed for details. Some familiarity with the basic approach is required to understand our new Active Appearance Model algorithm.

The models were generated by combining a model of face shape variation with a model of the appearance variations of a shape-normalised face. The models were trained on 400 face images, each labelled with 122 landmark points representing the positions of key features. The shape model was generated by representing each set of landmarks as a vector, $\mathbf{x}$ and applying a principal component analysis (PCA) to the data. Any example can then be approximated using:

$$\mathbf{x} = \bar{\mathbf{x}} + \mathbf{P}_s \mathbf{b}_s \tag{1}$$

where $\bar{\mathbf{x}}$ is the mean shape, $\mathbf{P}_s$ is a set of orthogonal *modes of variation* and $\mathbf{b}_s$ is a set of shape parameters. If each example image is warped so that its control points match the mean shape (using a triangulation algorithm) we can sample the grey level information $\mathbf{g}$ from this *shape-normalised* face patch. By applying PCA to this data we obtain a similar model:

$$\mathbf{g} = \bar{\mathbf{g}} + \mathbf{P}_g \mathbf{b}_g \tag{2}$$

The shape and appearance of any example can thus be summarised by the vectors $\mathbf{b}_s$ and $\mathbf{b}_g$. Since there are correlations between the shape and grey-level variations, we apply a further PCA to the concatenated vectors, to obtain a combined model of the form:

$$\mathbf{x} = \bar{\mathbf{x}} + \mathbf{Q}_s \mathbf{c} \tag{3}$$

$$\mathbf{g} = \bar{\mathbf{g}} + \mathbf{Q}_g \mathbf{c} \tag{4}$$

where $\mathbf{c}$ is a vector of *appearance* parameters controlling both the shape and grey-levels of the model, and $\mathbf{Q}_s$ and $\mathbf{Q}_g$ map the value of $\mathbf{c}$ to changes in the shape and shape-normalised grey-level data. A face can be synthesized for a given $\mathbf{c}$ by generating the shape-free grey-level image from the vector $\mathbf{g}$ and warping it using the control points described by $\mathbf{x}$ (see [3] for details).

The 400 examples lead to 23 shape parameters, $\mathbf{b}_s$, 114 grey-level parameters, $\mathbf{b}_g$, but only 80 combined appearance model parameters, $\mathbf{c}$ being required to explain 98% of the observed variation.

Figure 1 shows an unseen example image alongside the model reconstruction of the face patch (overlaid on the original image).

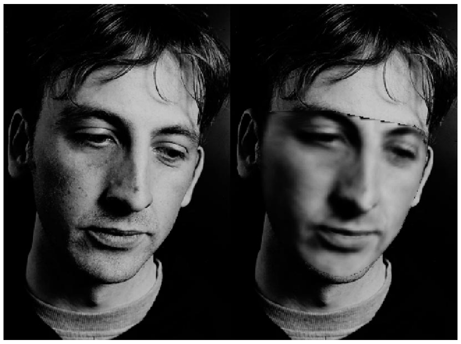

Figure 1. Example of combined model representation of an unseen image. Original image on left. Overlaid model reconstruction on right.

3 Generating Face Hypotheses

We adopt a two-stage strategy for matching the appearance model to face images. The first step is to find an approximate match using a simple and rapid approach. We assume no initial knowledge of where the face may lie in the image, or of it's scale and orientation. A simple eigen-face model[9] is used for this stage of the location. A correlation score, S, between the eigen-face representation of the image data, $\mathbf{M}$ and the image itself, $\mathbf{I}$ can be calculated at various scales, positions and orientations:

$$S = |\mathbf{I} - \mathbf{M}|^2 \tag{5}$$

Although in principle the image could be searched exhaustively, it is much more efficient to use a stochastic scheme similar to that of Matas *et al* [5]. We sub-sample both the model and image to calculate the correlation score using only a small fraction of the model sample points. Figure 2 shows typical face hypotheses generated using this method. The average time for location was around 0.2sec using 10% of the model sample points.

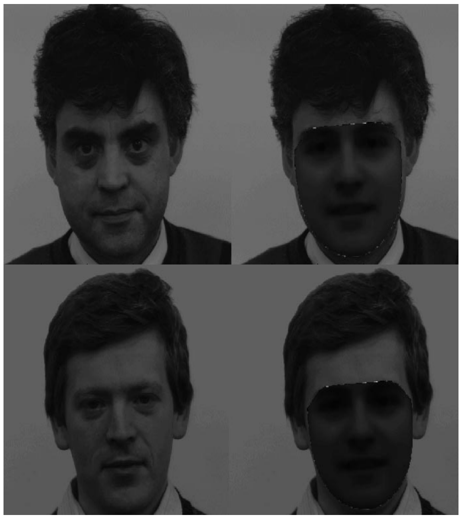

Figure 2. Example of generated face hypotheses. Average location time: 0.2sec at 10% sampling.

4 Active Appearance Model Search

We now address the central algorithm: given a full appearance model as described above and a reasonable starting approximation we propose a scheme for adjusting the model parameters efficiently, such that a synthetic face is generated, which matches the image as closely as possible. We first outline the basic idea, before giving details of the algorithm.

4.1 Overview of AAM Search

We wish to treat interpretation as an optimisation problem in which we minimise the difference between a real face image and one synthesised by the appearance model. A difference vector $\delta\mathbf{I}$ can be defined:

$$\delta\mathbf{I} = \mathbf{I_i} - \mathbf{I_m} \tag{6}$$

where $\mathbf{I_i}$ is the vector of grey-level values in the image, and $\mathbf{I_m}$, is the vector of grey-level values for the current model parameters:

To locate a best match between model and image, we wish to minimize the magnitude of the difference vector, $\Delta = |\delta \mathbf{I}|^2$, by varying the model parameters, $\mathbf{c}$.

Since the model has around 80 parameters, this appears at first to be a very difficult optimisation problem involving search in a very high-dimensional space. We note, however, that each attempt to match the model to a new face image, is actually a similar optimisation problem. We propose to learn something about how to solve this class of problems in advance. By providing a-priori knowledge of how to adjust the model parameters during during image search, we arrive at an efficient run-time algorithm. In particular, we might expect the spatial pattern in $\delta \mathbf{I}$, to encode information about how the model parameters should be changed in order to achieve a better fit. For example, if the largest differences between the model and the image occurred at the sides of the face, that would imply that a parameter that adjusted the width of the model face should be adjusted. This expected effect is seen in figure 3.

In adopting this approach there are two parts to the problem: learning the relationship between $\delta \mathbf{I}$ and the error in the model parameters, $\delta \mathbf{c}$ and using this knowledge in an iterative algorithm for minimising Δ.

4.2 Learning to Correct Model Parameters

The simplest model we could choose for the relationship between $\delta \mathbf{I}$ and the error in the model parameters (and thus the correction which needs to be made) is linear:

$$\delta \mathbf{c} = \mathbf{A} \delta \mathbf{I} \tag{7}$$

This turns out to be a good enough approximation to provide good results. To find $\mathbf{A}$, we perform multiple multivariate linear regression on a large sample of known model displacements, $\delta \mathbf{c}$, and the corresponding difference images, $\delta \mathbf{I}$. We can generate these large sets of random displacements, by perturbing the 'true' model parameters for the images in the training set by a known amount. As well as pertubations in the model parameters, we also model small displacements in 2D position, scale, and orientation. These extra 4 parameters are included in the regression; for simplicity of notation, they can, however, be regarded simply as extra elements of the vector $\delta \mathbf{c}$. In order to obtain a well-behaved relationship it is important to choose carefully the frame of reference in which the image difference is calculated. The most suitable frame of reference is the shape-normalised face patch described in section 2. We calculate a difference thus: for the current location of the model, calculate the *image* grey-level sample vector, $\mathbf{g_i}$, by warping the image data at the current location into the shape-normalised face patch. This is compared with the *model* grey-level sample vector, $\mathbf{g_m}$, calculated using equation 4:

$$\delta \mathbf{g} = \mathbf{g_i} - \mathbf{g_m} \tag{8}$$

Thus, we can modify equation 7:

$$\delta \mathbf{c} = \mathbf{A} \delta \mathbf{g} \tag{9}$$

The best range of values of $\delta \mathbf{c}$ to use during training is determined experimentally. Ideally we seek to model a relationship that holds over as large a range errors, $\delta \mathbf{g}$ as possible. However, the real relationship is found to be linear only over a limited range of values In our experiments, the model used 80 parameters. The optimum pertubation level was found to be around 0.5 standard deviations (over the training set) for each model parameter. Each parameter was perturbed from the mean by a value between 0 and 1 standard deviation. The scale, angle and position were perturbed by values ranging from 0 to +/- 10% (positional displacements are relative to the face width.) After performing linear regression, we can calculate an R^2 statistic for each parameter perturbation, $\delta \mathbf{c_i}$ to measure how well the displacement is 'predicted' by the error vector $\delta \mathbf{g}$. The average R^2 value for the 80 parameters was 0.82, with a maximum of 0.98 (the 1st parameter) and a minimum of 0.48. Figure 3 illustrates the shape-free error image reconstructed for $\delta \mathbf{g}$, for a deviation of 2 standard deviations in the 1st model parameter, and a horizontal displacement of 10 pixels.

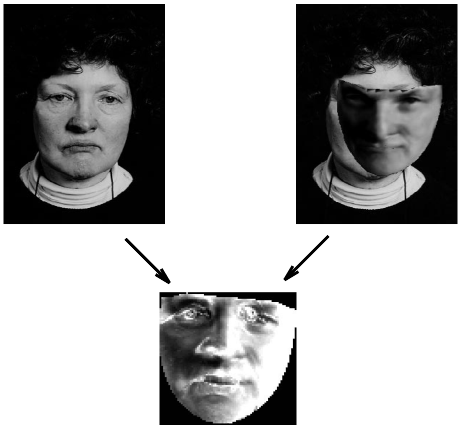

Figure 3. Shape-free error image. Top left: original, Top right: pertubed model placement, Bottom centre: Shape-normalised difference image

4.3 Iterative Model Refinement

Given a method for predicting the correction which needs to made in the model parameters we can construct an iterative method for solving our optimisation problem. For a given model projection into the image, $\mathbf{c}$, we calcuate the grey-level sample error vector, $\delta\mathbf{g}$, and update the model estimate thus:

$$\mathbf{c}' = \mathbf{c} - \mathbf{A}\delta\mathbf{g} \tag{10}$$

If the initial approximation is far from the correct solution the predicted model parameters at the first iteration will generally not be very accurate but should reduce the energy in the difference image. This can be ensured by scaling $\mathbf{A}$ so that the prediction reduces the magnitude of the difference vector, $|\delta\mathbf{g}|^2$, for all the examples in the training set. Given the improved value of the model parameters, the prediction made in the next iteration should be better. The procedure is iterated to convergence. Typically the algorithm converges in around 5-10 iterations from fairly poor starting approximations - more quantitative data are given in the results section.

5 Experimental Results

The method was tested on a set of 80 previously unseen face images. Figure 4 shows three example images used for testing and the 'true' model reconstruction, based on hand-annotation of the face location and shape.

Figure 5 illustrates the result of applying AAM search to these images. The left hand image shows the original overlaid with the initial hypothesis for face location. In practise, we usually have better starting hypotheses than shown here, however, in order to illustrate the convergence properties of AAM search, we have deliberately displaced the hypotheses generated by the stochastic generator, so as to make the problem 'harder'. Alongside the initial approximation are shown the search result afters iterations 1,5 and 12, respectively.

5.1 Reconstruction Error

We tested the reconstruction error of AAM search over a test set of 80 unseen images. The reconstruction error for each image is calculated as the magnitude of the shape-normalised grey-level sample vector, $|\delta\mathbf{g}|^2$. Figure 6 show a graph of reconstruction error versus iteration:

Two plots are shown: The solid curve is a plot of average error versus iteration for the test set. The dashed curve shows the worst case encountered in the test. The two horizontal lines indicate the error measured when the model is fitted using accurate, hand-labelled points, for the average and worst case respectively. The error is measured in average grey-level difference per sample pixel, where pixels take a value from 0 to 63.

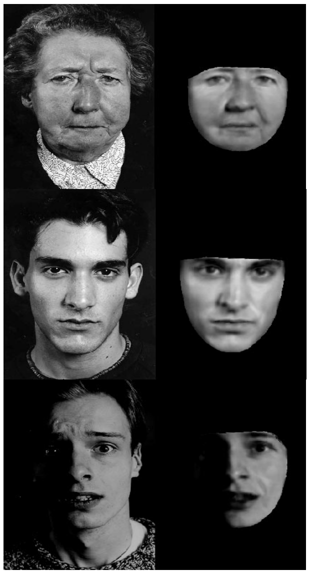

Figure 4. Example test images with 'true' reconstruction, based on hand annotation

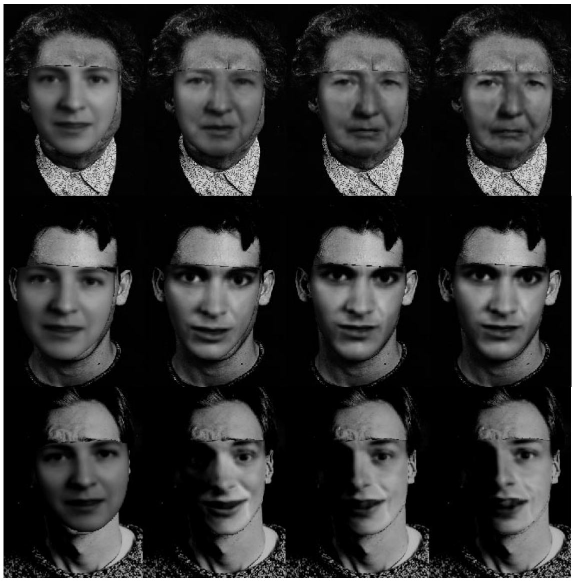

Figure 5. Search results: Initial location, Iteration 2, Iteration 5, Iteration 12

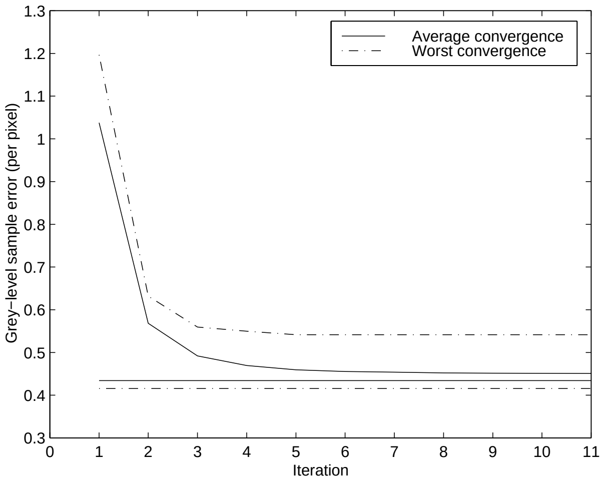

Figure 6. Search Convergence :The solid curve is a plot of average error versus iteration for the test set. The dashed curve shows the worst case encountered in the test. The two horizontal lines indicate the error measured when the model is fitted using accurate, hand-labelled points, for the average and worst case respectively. The error is measured in average grey-level difference per sample pixel, where pixels take a value from 0 to 63.

6 Discussion and Conclusions

We have demonstrated an iterative scheme for fitting an Active Appearance Model to face images. The method makes use of learned correlation between model-displacement and the resulting difference image. Given a reasonable initial starting position, the search converges quickly, and is comparable in speed to an Active Shape Model. Using AAMs real-time tracking should be possible on a standard PC. However, since all the image evidence is used, the procedure is more robust than ASM search alone. We are currently investigating further efficiency improvements, for example, subsampling both model and image, as was used in the method for hypotheses generation. It is intended to use AAM search to track faces in sequences, using the tracking scheme of Edwards *et al* [3]. This scheme requires both off-line and on-line 'de-coupling' of sources of variation due to ID,Pose,Lighting and Expression. The decoupling makes use of the full appearance model and thus provides more information when used with full AAM search than with ASM search alone. The dynamic constraints and evidence integration of the tracking scheme provide further robustness and thus we expect excellent performance from a full AAM tracking scheme.

References

[1] M. J. Black and Y. Yacoob. Recognizing Facial Expressions under Rigid and Non-Rigid Facial Motions. In *International Workshop on Automatic Face and Gesture Recognition 1995*, pages 12–17, Zurich, 1995.

[2] M. Covell. Eigen-points: Control-point Location using Principal Component Analysis. In *International Workshop on Automatic Face and Gesture Recognition 1996*, pages 122–127, Killington, USA, 1996.

[3] G. J. Edwards, C. J. Taylor, and T. Cootes. Learning to Identify and Track Faces in Image Sequences. In *British Machine Vision Conference 1997*, Colchester, UK, 1997.

[4] T. Ezzat and T. Poggio. Facial Analysis and Synthesis Using Image-Based Models. In *International Workshop on Automatic Face and Gesture Recognition 1996*, pages 116–121, Killington, Vermont, 1996.

[5] K. J. J. Matas and J. Kittler. Fast Face Localisation and Verification. In *British Machine Vision Conference 1997*, Colchester, UK, 1997.

[6] A. Lanitis, C. Taylor, and T. Cootes. Automatic Interpretation and Coding of Face Images Using Flexible Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):743–756, 1997.

[7] C. Nastar, B. Moghaddam, and A. Pentland. Generalized Image Matching: Statistical Learning of Physically-Based Deformations. In 4^{th} *European Conference on Computer Vision*, volume 1, pages 589–598, Cambridge, UK, 1996.

[8] S. Sclaroff and J. Isidoro. Active Blobs. In 6^{th} *International Conference on Computer Vision*, pages 1146–1153, Mumbai, India, 1998.

[9] M. Turk and A. Pentland. Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[10] P. Viola and W. W. III. Alignment by Maximization of Mutual Information. In 5^{th} *International Conference on Computer Vision*, pages 16–23, Cambridge, USA, 1995.