BMVC99

Automatically Building Appearance Models from Image Sequences using Salient Features.

K.N.Walker, T.F.Cootes and C.J.Taylor
Dept. Medical Biophysics,
Manchester University, UK
Tel: +44 (0)161 275 5130
Fax: +44 (0)161 275 5145
email: knw@sv1.smb.man.ac.uk

Abstract

We address the problem of automatically placing landmarks across an image sequence to define correspondences between frames. The marked up sequence is then used to build a statistical model of the appearance of the object within the sequence. We locate the most salient object features from within the first frame and attempt to track them throughout the sequence. Salient features are those which have a low probability of being mis-classified as any other feature, and are therefore more likely to be robustly tracked throughout the sequence. The method automatically builds statistical models of the objects shape and the salient features appearance as the sequence is tracked. These models are used in subsequent frames to further improve the probability of finding accurate matches. Results are shown for several face image sequences. The quality of the model is comparable with that generated from hand labelled images.

1 Introduction

Statistical models of shape and appearance have proved powerful tools for interpreting images, particularly when combined with algorithms to match the models to new images rapidly [5, 4, 8]. In order to construct such models we require sets of labelled training images. The labels consist of landmark points defining the correspondences between similar structures in each image across the set.

The most time consuming and scientifically unsatisfactory part of building the models is the labeling of the training images. Manually placing hundreds of points on every image is both tedious and error prone. To reduce the burden, semi-automatic systems have been developed. In these a model is built from the current set of examples (possibly with extra artificial modes included in the early stages) and used to search the next image. The user can edit the result where necessary, then add the example to the training set. Though this can considerably reduce the time and effort required, labelling large sets of images is still difficult.

We present a method which, given a sequence of an object and its outline in the first frame, automatically returns a set of correspondences across the entire sequence. We take advantage of the fact that it is easier to track correspondences across a sequence than find

them between two arbitrary images from a training set. The found correspondences can then be used to build an appearance model of the object.

The local image structure at each image point is described by vectors of Gaussian partial derivatives (or *feature vectors*) computed at a range of scales. We can calculate the spatial positions and scales of the salient features within the first frame by defining salient feature vectors [14, 13] to be the ones which lie in the low density regions of the distribution of all feature vectors. Salient features have a low probability of being mistaken for any other feature within the object. They therefore have a low probability of generating false positive matches when searched for in subsequent frames.

As the sequence is searched, we build statistical models of each salient feature. These models enable us to use all the information gained from searching previous frames to help find the correct matches in subsequent frames. We also build a model of the relative positions of each feature and ensure that only sensible configurations of features are selected.

In the following we briefly review how to construct appearance models. We describe how salient features are located in the first frame, and how they are used to derive correspondences across an image sequence. Finally we show some examples of automatically built appearance models and compare them with appearance models built from manually labelled training data.

2 Background

2.1 Saliency and Automatic Landmarking

Many authors have shown that using the saliency of image features can improve the robustness in object recognition algorithms [3, 12], but this typically been applied to finding salient segments on an object boundary.

Approaches to automatic landmark placement in 2D have assumed that contours have already been segmented from the training sets [7, 1]. This is a process in its self can time consuming and involve human intervention.

Baumberg and Hogg [1] describe a system which generates landmarks automatically for outlines of walking people. While this process is satisfactory for silhouettes of pedestrians, it is unlikely that it will be generally successful. The authors went on to describe how the position of the landmarks can be iteratively updated in order to generate improved shape models generated from the landmarks [2].

Many people have used jets of Gabor Wavelets or Gaussian Partial Derivative to describe and locate features in new image examples [9, 10]. Typically the location of the features modelled are selected by hand, choosing for example, heavily textured areas. We believe that there is an optimum set of these features which best determine the object, and attempting to select them manually risks compromising system performance.

2.2 Appearance Models

An appearance model can represent both the shape and texture variability seen in a training set. The training set consists of labelled images, where key landmark points are marked on each example object.

Given such a set we can generate a statistical model of shape variation by applying Principal Component Analysis (PCA) to the set of vectors describing the shapes in the training set (see [5] for details). The labelled points, $\mathbf{x}$, on a single object describe the shape of that object. 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 vector, $\mathbf{P}_s$ is a set of orthogonal *modes of shape variation* and $\mathbf{b}_s$ is a vector of shape parameters.

To build a statistical model of the grey-level appearance we warp each example image so that its control points match the mean shape (using a triangulation algorithm). Figure 1 shows three examples of labelled faces. We then sample the intensity information from the *shape-normalised* image over the region covered by the mean shape. To minimise the effect of global lighting variation, we normalise the resulting samples.

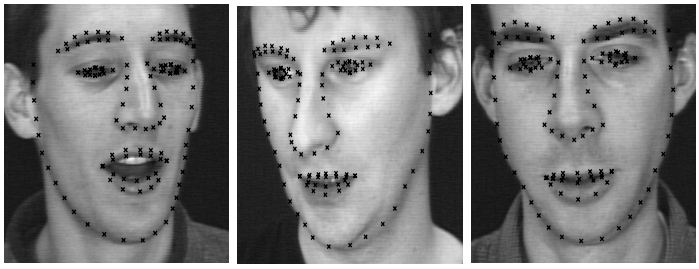


Figure 1: Examples of faces labelled with consistent landmarks

By applying PCA to the normalised data we obtain a linear model:

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

where $\bar{\mathbf{g}}$ is the mean normalised grey-level vector, $\mathbf{P}_g$ is a set of orthogonal *modes of intensity variation* and $\mathbf{b}_g$ is a set of grey-level parameters.

The shape and texture are correlated. By further analysis [4] we can derive a joint model of shape and texture:

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

$$\mathbf{y} = \bar{\mathbf{y}} + \mathbf{Q}_y \mathbf{c} \tag{4}$$

where $\mathbf{Q}_x$ and $\mathbf{Q}_y$ represent modes of shape and texture variation, and $\mathbf{c}$ is a vector of appearance model parameters.

An example image 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 $\mathbf{x}$.

3 Automatically Building Appearance Models

To build an Appearance Model it is necessary to calculate a dense correspondence between all frames in the training set. This is typically achieved by placing consistent landmarks on all training frames and using a triangulation algorithm to approximate the dense

correspondence. We have attempted to find a consistent set of landmarks automatically by selecting salient features in the first image in the sequence. These are the features which have the lowest probability of being mistaken with any other features in that image. By searching for these features in subsequent images we minimize the probability of false positives. Robustness can further be improved by applying shape constraints learnt from previous frames to searches in subsequent frames.

We begin by explaining how salient features are chosen from the initial frame.

3.1 Locating salient features in the first image

The aim is to locate salient features, those which are most likely to be relocated correctly in a subsequent frame. Given only one example of the object, the best we can do is to attempt to find those features which are significantly different to all other features in the object. Ideally, these features would occur exactly once in each example of the object.

For every pixel within the object boundary we construct several feature vectors, each of which describes a feature centered on the pixel at a range of scales. In the following the feature vectors used were the first and second order normalised Gaussian partial derivatives, giving a five dimensional feature vector for each scale considered. Higher orders could be used depending on the computational power available. The full set of vectors describing all object features (one per image pixel), forms a multi-variate distribution in a *feature space*. By modeling the density in feature space we can estimate how likely a given feature is to be confused with other features. Salient features lie in low density areas of feature space.

In the rest of this section we describe how we modeled the density of feature space and how this can be used to select salient features.See [14, 13] for a more detailed description.

3.1.1 Modeling the Density of Feature Space

We estimate the local density $\hat{p}$ at point $\mathbf{x}$ in a feature space by summing the contribution from n Gaussian kernels, one centered on each sample, $\mathbf{x}_i$, in the distribution.

$$\hat{p}(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} G(\mathbf{x} - \mathbf{x}_i; \mathbf{\Sigma}) \tag{5}$$

where $G(\mathbf{x}; \mathbf{\Sigma})$ gives the probability of $\mathbf{x}$ being part of the normalised multi-variant Gaussian distribution, with covariance $\mathbf{\Sigma}$ and a mean of 0, $\mathbf{\Sigma} = h\mathbf{S}$ where $\mathbf{S}$ is the covariance of the entire distribution and h is a scaling factor.

This method of modeling the density of feature space gives very good results but at a cost of a complexity of n^2. An approximation to this method is to model the distribution with less than n Gaussian Kernels [14].

3.1.2 Selecting Salient Features

The density estimate for each feature vector $\mathbf{v}(\sigma)$ corresponds directly to the saliency of the feature at scale σ. The lower the density, the more salient the point.

A saliency image can be constructed for each scale analysed by setting each pixel to the probability density at the corresponding feature vector at that scale. This forms a

saliency image stack. The positions and scales of the most salient features are then found by locating the lowest troughs in the saliency images. To ensure that the features selected are distributed evenly across the object, the following algorithm is used:

REPEAT
  Select the next lowest trough in the saliency image stack
  IF trough is more than 3σ pixel's from any previously selected trough
    Mark trough as a representing a salient feature
  ELSE
    Discard trough
  ENDIF
UNTIL no more trough's

Figure 2 shows a saliency image stack calculated at 3 scales from an image of a face. The features selected are shown on the original image, the circles around each feature indicate the scale of the features (i.e. the region filtered in order to construct the feature vector).

The result of this analysis is n salient features which are described by $\mathbf{v}_{1i}$, s_i and $\mathbf{x}_{1i}$, where $\mathbf{v}_{ji}$ is the feature vector of salient feature i in frame j, s_i is the scale at which feature i was found to be salient, and $\mathbf{x}_{ji} = (x_{ji}, y_{ji})$ are the x and y coordinates of feature i in frame j respectively.

3.2 Locating the features in the jth frame

When locating the features in the jth frame we make the following two assumptions:

- The features will not move more than a given number, r, pixels between frames.
- The scale of the object and therefore features will not change significantly.

These assumptions help constrain the search and reduce processing time. Only πr^2 candidate pixels in each frame need to be considered in order to locate the best match for one particular feature. A candidate pixel has an associated feature vector, $\mathbf{v}$. We assume the quality of match to the i^{th} feature model using $M_i(\mathbf{v})$;

$$M_i(\mathbf{v}) \quad = \quad (\mathbf{v} - \bar{\mathbf{v}}_i)^T \mathbf{S}^{-1} (\mathbf{v} - \bar{\mathbf{v}}_i) \tag{6}$$

where $\bar{\mathbf{v}}$ is the mean of the matches for the i_{th} feature, $\bar{\mathbf{v}}_i = \frac{1}{j-1} \sum_{k=1}^{j-1} \mathbf{v}_{ki}$, and

$$\mathbf{S} = \begin{cases} \mathbf{C}_{s_i} & \text{if } j = 2 \\ \\ \frac{1}{j}(\sum_{k=1}^{j-1} (\mathbf{v}_k i - \bar{\mathbf{v}}_i)^2 + \lambda_r \mathbf{I}) & \text{if } j > 2 \end{cases}$$

C_σ is the covariance matrix of all the features at scale σ within the object boundary from frame one and $\lambda_r \mathbf{I}$ is a regularising term to avoid singular matrix inversion. The smaller $M_i(\mathbf{v})$, the better the match. $M_i(\mathbf{v})$ is linearly related to the log of the probability that $\mathbf{v}$ comes from the distribution of feature model i.

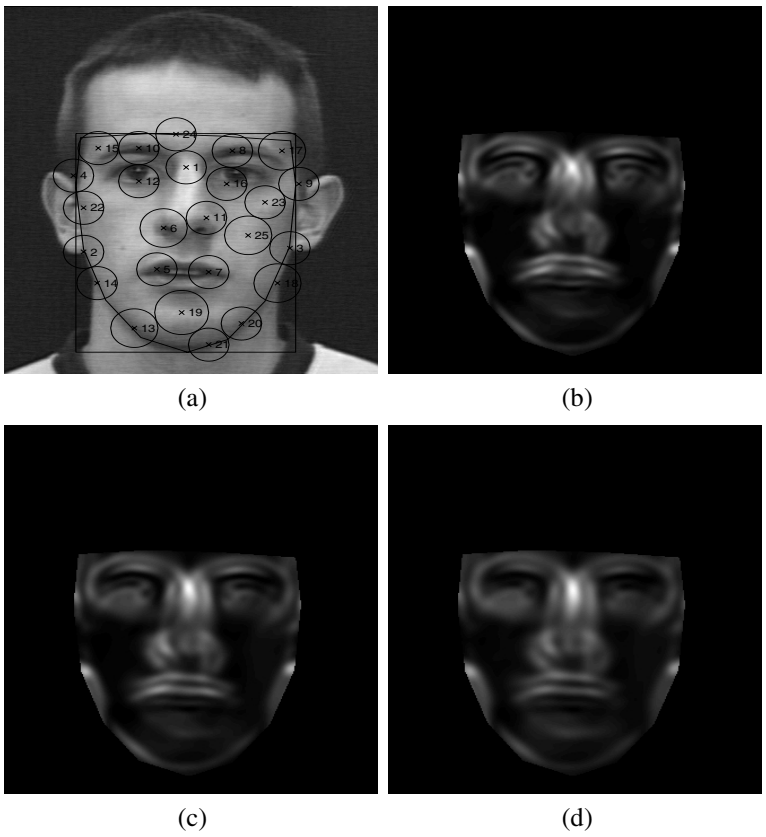
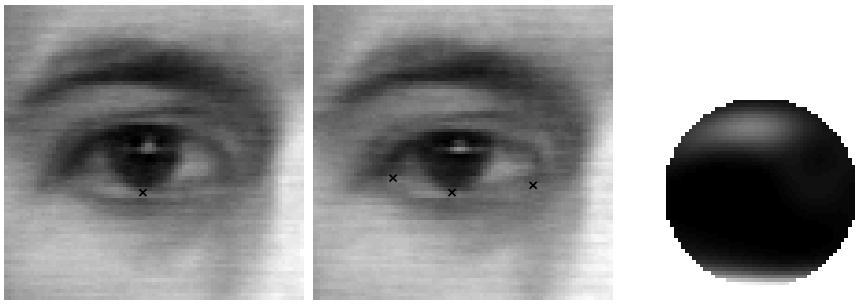


Figure 2: Images (b), (c) and (d) are saliency images extracted from image (a) at scales 6, 7 and 8 standard deviations respectively. The circles on image (a) show the locations of the salient features selected. The size of the circles represent the image region that the salient feature represents (or the scale).

By calculating $M_i(\mathbf{v})$ for all candidates for a particular feature, i, we can form a similarity image. This is done by plotting $M_i(\mathbf{v})$ in the image domain. We locate the m best matches for the feature by locating the m lowest trough's in the similarity image. Figure 3 shows an example of this.

The shape of an object is represented by n salient features. We have selected m possible candidate matches for each salient feature. By selecting one candidate match for each salient feature we form an *object hypothesis*. Generating all possible combinations would result in m^n object hypotheses. In order to reduce the number of hypotheses, candidates which correspond to the largest similarity values, $M_i(\mathbf{v})$, can be discarded (i.e. remove the most improbable matches). The number of matches discarded depends on the computational power available.

Each hypothesis is given a probability of being correct, based on the evidence obtained from previous frames. We define the probability of a hypothesis being correct to be equal to the probability of all features being correct together with the probability of the shape



(a) (b) (c)

Figure 3: Calculating a feature match in subsequent frames. (c) is the similarity image obtained whilst searching for the feature in (a) in the frame shown in (b). The trough's in (c) represent the matches, these are highlighted in (b)

formed by the salient features being correct. The shape formed by the n salient features, for a particular frame, j, is described by a $2n$ element *shape vector*, $\mathbf{x}_j$, where:

$$\mathbf{x}_j = (x_{j1}, ..., x_{jn}, y_{j1}, ..., y_{jn})^T \tag{7}$$

The quality of fit of a shape vector, $\mathbf{x}_j$, in frame, j, is given by:

$$M(\mathbf{x}_j) \quad = \quad (\mathbf{x}_j - \bar{\mathbf{x}})^T \mathbf{H}^{-1} (\mathbf{x}_j - \bar{\mathbf{x}}) \tag{8}$$

where $\bar{\mathbf{x}}$ and $\mathbf{H}$ are the mean and covariance matrix of shape vectors from frame 1 to $j-1$ after they have been aligned using Procustes Analysis [6]. If $j = 2$ then $H = \mathbf{I}\sigma$ where $\mathbf{I}$ is a $(2n\text{x}2n)$ identity matrix and σ is an estimate of the variance of the features between frames in pixel's. $M(\mathbf{x}_j)$ is also linearly related to the log probability that $\mathbf{x}_j$ is drawn from the distribution shape vectors from frames 1 to $j-1$.

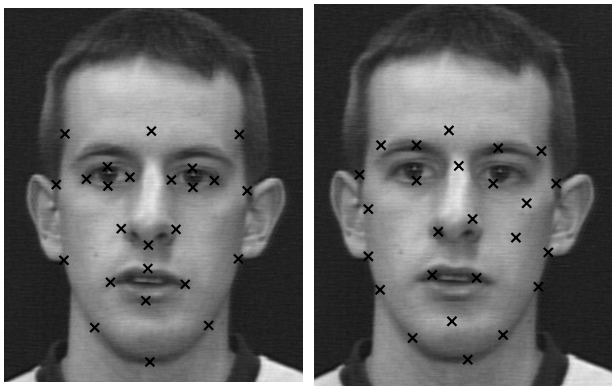
We assume features can be treated as independent of one another and the global shape. A measure of the quality of a particular hypothesis, $M(h_k)$, is thus given by:

$$M(h_k) = M(\mathbf{x}_{h_k}) + \sum_{i=1}^{n} M_i(\mathbf{v}_{h_k i}) \tag{9}$$

where $\mathbf{x}_{h_k}$ is the shape vector for hypothesis h_k and $\mathbf{v}_{h_k i}$ is the feature vector that hypothesis h_k has selected to describe feature i. We select the hypothesis with the smallest $M(h_k)$ as the most likely solution.

4 Results

The automatic method gives qualitatively good results if the sequence is correctly tracked. Figure 5(a) illustrates the modes of variation for an automatically trained model. As a



(a) (b)

Figure 4: Landmarks chosen manually (a) and automatically (b).

comparison we also build two other models. The first trained using n hand placed landmarks, the second trained using only 4 landmarks placed on each corner of the smallest rectangle which could contain the face in each frame, essentially an eigenface [11]. This second model was used to examine the affect on the texture error if the texture and shape information where not modeled separately (no normalization of shape). Figure 4 shows the positions for the landmarks for the hand placed model (a) and the automatic model (b). Figure 5(b) shows the modes for the manually trained model. They appear very similar to those for the automatically trained model.

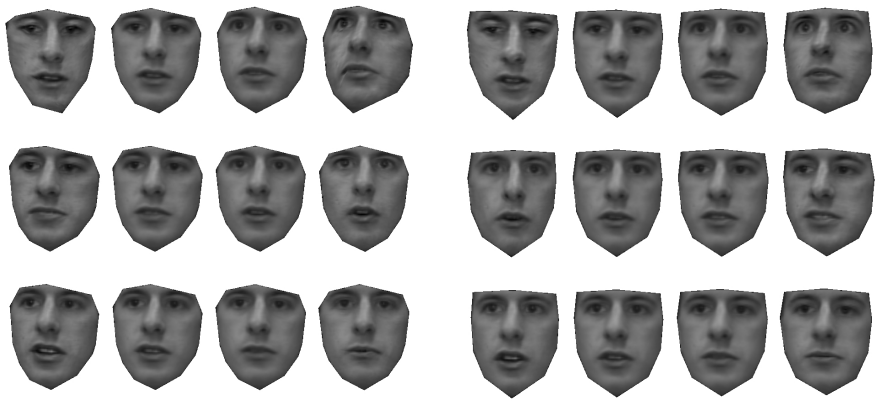
It is difficult to assess the results quantitatively without considering the application to which the model will be put. For effective coding (compression), we require that truncating the number of models modes has a minimal effect on the reconstruction error. We can thus compare models by how the reconstruction error varies with the number of modes.

Figure 6 shows how the reconstruction texture error, (a), and the shape error, (b), increase as the model modes decrease. The graphs show that both the texture error and shape error are reduced by training the model automatically. The automatic method thus leads to a more compact model.

5 Discussion

We have demonstrated one possible approach to automatically training appearance models. The system both locates the most suitable features to use as model points, and tracks their correspondence across frames. We have shown that the method can be used to automatically make models of the human face for image sequences.

However, if features move and change shape significantly between frames, false correspondences can result. Also in long sequences, features which were salient in the first frame are not necessarily still salient in the 50th frame. Walker *et al* [15] proposed a method of calculating the locations of salient features over a number of frames given an approximate correspondence. This idea could be incorporated into our scheme. When



(a) (b)

Figure 5: Models of variation for an automatically trained (a) and manually trained (b) model.

searching frame j, instead of locating features which were salient in frame 1, we could locate features which were salient in frames $j-1$ to, say, $j-5$. However, this would require additional work (eg using interpolation between points) to compute correspondences across the entire sequence.

The results presented in this paper have used a relatively small number of landmarks (25). We have observed that the coarse scale features are, in general, much more reliable than the fine scale features. We would like to extend this scheme to a find a greater number of correspondences by adopting a multiscale approach. Coarse scale salient features would be found first, then the resulting coarse scale correspondence could be used to constrain a search for finer scale salient features. This would lead to a more accurate overall correspondence.

Our method would be well suited to in the animation industry for building models of faces or other deformable objects with the intention of using the models to produce photo realistic animations.

References

[1] Adam Baumberg and David Hogg. Learning flexible models from image sequences. In 3^{nd} *European Conference on Computer Vision*, pages 299–308, 1994.

[2] Adam Baumberg and David Hogg. An adaptive eigenshape model. In David Pycock, editor, 6^{th} *British Machine Vison Conference*, pages 87–96. BMVA Press, September 1995.

[3] R. C. Bolles and R. A. Cain. Recognising and locating partially visible objects: the local-feature-focus method. *Int. J. Robotics Res*, 1:57–82, 1982.

[4] T.F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. In H.Burkhardt and B. Neumann, editors, 5^{th} *European Conference on Computer Vision*, volume 2, pages 484–498. Springer, 1998.

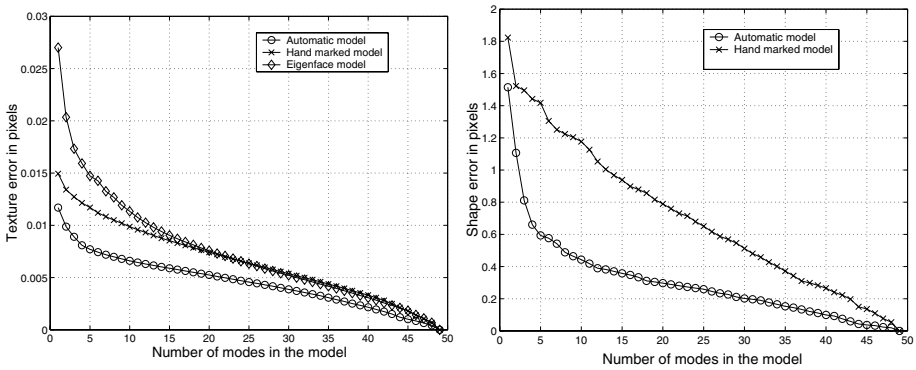


Figure 6: Compares the texture error (a) and shape error (b) of an automatically built model with that of an eigen face model and a model trained using hand placed landmarks.

[5] Tim F. Cootes, Christopher J. Taylor, David H. Cooper, and Jim Graham. Active shape models - their training and application. *Computer Vision and Image Understanding*, 61(1):38–59, January 1995.

[6] C. Goodall. Procrustes methods in the statistical analysis of shape. *Journal of the Royal Statistical Society B*, 53(2):285–339, 1991.

[7] Andrew Hill and Christopher J. Taylor. Automatic landmark identification using a new method of non-rigid correspondence. In 15^{th} *Conference on Information Processing in Medical Imaging*, pages 483–488, 1997.

[8] A. Lanitis, C.J. Taylor, and T.F. Cootes. Automatic interpretation and coding of face images using flexible models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):743–756, 1997.

[9] Micheal Lyons and Shigeru Akamatsu. Coding Facial Expressions with Gabor Wavelets. In 3^{rd} *International Conference on Automatic Face andGesture Recognition 1998*, pages 200–205, Nara, Japan, 1998.

[10] Hai Tao, Ricardo Lopez, and Thomas Huang. Tracking Facial Features Using Probabilistic Network. In 3^{rd} *International Conference on Automatic Face andGesture Recognition 1998*, pages 166–170, Nara, Japan, 1998.

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

[12] J. L. Turney, T. N. Mudge, and R. A. Voltz. Recognising partially occluded parts. *IEEE Trans. PAMI*, 7:410–421, 1985.

[13] K. N. Walker, T. F. Cootes, and C. J. Taylor. Locating salient facial features using image invariants. In 3^{rd} *International Conference on Automatic Face andGesture Recognition 1998*, pages 242–247, Nara, Japan, 1998.

[14] K. N. Walker, T.F. Cootes, , and C. J. Taylor. Correspondence based on distinct points using image invariants. In 8^{th} *British Machine Vison Conference*, pages 540–549, Colchester, UK, 1997.

[15] K. N. Walker, T.F. Cootes, , and C. J. Taylor. Locating salient object features. In P.H. Lewis and M.S. Nixon, editors, 9^{th} *British Machine Vison Conference*, volume 2, pages 557–566, Southampton, UK, September 1998. BMVA Press.