Face Tracking and Pose Representation*

Stephen McKenna, Shaogang Gong and J. J. Collins
Machine Vision Lab., Dept. of Computer Science
Queen Mary and Westfield College, Mile End Rd., London.
E-mail: stephen@dcs.qmw.ac.uk

Abstract

We describe a dynamic face tracking system based on an integrated motion-based object tracking and model-based face detection framework. The motion-based tracker focuses attention for the face detector whilst the latter aids the tracking process. The system produces segmented face sequences from complex scenes with poor viewing conditions in surveillance applications. We also investigate a Gabor wavelet transform as a representation scheme for capturing head rotations in depth. Principal components analysis was used to visualise the manifolds described by pose changes. Qualitative results are given.

1 Introduction

In order to analyse and recognise peoples' faces in a realistically unconstrained environment, robust tracking and segmentation are needed to provide sequences of normalised face images. Although such a *normalisation process* is often treated as a separate preprocessing step, it is an inherent part of face recognition. The ability of a system to produce normalised face sequences implies that it recognises faces as a unique class of objects and in a manner which exhibits invariance under many possible transformations. These transformations include changes in illumination, orientation and position in 3D space relative to the camera, and changes in motion. Perhaps most problematic are rotations in depth ("pose" changes).

There are two broad approaches to the problem of tracking moving objects. Motion-based approaches depend on a robust method for grouping visual motions consistently over time [15]. They tend to be fast but do not guarantee that the tracked regions have any meaning [14]. Model-based approaches, on the other hand, can impose high-level semantic knowledge more readily but suffer from being computationally expensive due to the need to cope with scaling, translation, rotation and deformation. We propose a face tracking system which combines motion-based and model-based representations. We suggest an integrated detection-tracking system with a *closed-loop* in which a motion-based tracker reduces the search space for a model-based face detection whilst the latter aids the motion tracking and helps resolve ambiguities which arise in the grouping of visual

* Research funded by EPSRC Grant GR/K44657 and EC Grant CHRX-CT94-0636.

BMVC 1996 doi:10.5244/C.10.31

motion. Our system outputs segmented face sequences suitable for face recognition from scenes containing several people[1]. The availability of face sequences also allows the use of temporal information to constrain the recognition task [26].

Face models can be roughly categorised as explicitly 3D [2], 2D geometric feature-based [10], and 2D appearance-based [19, 27]. We subscribe to the view that the appearance-based approach is more promising whilst neither 3D models nor 2D geometric features can be easily extracted and matched robustly under changing viewing conditions. Face representation based on specific features for all poses may be difficult to find since different features seem relevant at different poses[2]. A more plausible [4] and robust [3] approach requires extraction of pose relevant information in a manner somewhat holistic and not dependent on specific features. This does not mean exhaustive representation. Appearance-based face recognition need not require every view of every person to be stored [1, 4]. The pose sphere could be represented by a few views. It is, however, unclear what image representation gives the most easily measurable pose distribution of faces.

Principal components analysis (PCA) is widely used for reducing dimensionality to enable efficient matching [19]. However, faces represented by principal components (PC's) are sensitive to illumination, scale, translation and rotation [9, 24]. Murase and Nayar [23] have used the PC's of many views of a single object to visualise the low-dimensional manifold described by changes due to rotation in depth and illumination conditions. The object's pose could then be determined by its position on this manifold. In the second part of this paper we use PCA in a similar manner to investigate the distribution of face pose in high-dimensional representation spaces. In particular, we use a Gabor wavelet transform (GWT) to investigate the role of locally oriented features at a range of spatial frequencies in selecting face pose[3]. This can also be regarded as part of the normalisation process and provides invariance under scaling as well as changes in illumination conditions, skin tone and hair colour.

The remainder of this paper is arranged as follows. Section 2 describes a motion-based tracking system. Section 3 briefly reviews face detection in static scenes and describes the method used by our system for dynamic scenes. Section 4 discusses system integration. In section 5 we turn our attention to face representation and outline a GWT scheme and parametric pose eigenspaces. Qualitative results are presented. Finally, in section 6 we draw some conclusions.

2 Tracking multiple motions

In surveillance we would like to track the motion of several people against complex backgrounds using a stationary monochrome camera. This section describes such a tracker comprised of motion detection, grouping, matching and Kalman filters.

[1] In applications such as HCI and access control, face detection and tracking are often less demanding (e.g.[25]). There is usually a single user whose face fills much of the field of view at high resolution. In contrast, surveillance must deal with poor resolution and multiple moving objects [8], only some of which may have visible faces.

[2] For example, a silhouette helps distinguish between 3/4 and profile views but is not of much relevance in distinguishing between frontal and 3/4 views.

[3] We prefer GWT to Gaussian derivative filters [27] since the formulation is more unified and allows separation of magnitude and phase.

Visual motion can be best estimated at moving image contours where such estimates are likely to be most relevant and reliable [7, 13]. This can be effectively achieved by convolving the intensity "history" of each pixel $I(x, y, t)$ with the second order temporal derivative of a Gaussian function $G(t)$ yielding an image of temporal zero-crossings $S(x, y, t)$:

$$S(x, y, t) = \frac{\partial^2 G(t)}{\partial t^2} * I(x, y, t)$$

The motion of an "edge" produces a zero-crossing in $S(x, y, t)$ at the location of the edge in the middle frame of the "history" used for the temporal convolution [13]. Global illumination changes and changes in intensity levels of static objects do not result in such temporal zero-crossings. Normal components of visual motion can be estimated from the partial derivatives of $S(x, y, t)$ [7, 13]. Figure 1 shows an image from a sequence along with its temporally filtered image and the detected temporal zero-crossings.

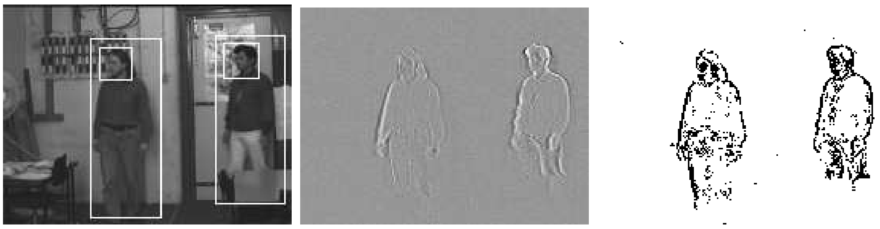

Figure 1: Motion-based tracking. Left: bounding boxes for tracked people and their heads. Centre: temporal convolution. Right: temporal zero-crossings.

The detected zero-crossings clustered. Each cluster should correspond to one object although in practice objects occasionally split or merge. In the current implementation, clustering uses only Euclidean distances between zero-crossings. Each cluster is modelled by the mean and variance of its zero-crossings in the directions of the image axes. A temporally consistent cluster list is maintained using time-symmetric matching [31]. Forward matching selects the nearest cluster in the current frame whilst reverse matching selects the oldest candidate cluster. Clusters are thus consistently tracked even if they sometimes erroneously split apart. Given a sufficiently high ratio of frame rate to image-plane velocity, normal components of visual motion can be used to improve clustering and in particular to help segment occluding objects with differing visual motions.

Kalman filters are used to track clusters robustly based upon measurements of position, motion and shape [16, 20, 34]. After a cluster has been tracked for 4 frames its bounding box is initialised. It is then assigned a "persistence", p, and will be maintained even in the absence of a matching cluster for up to p frames[4]. This allows objects to be tracked for short periods despite clustering errors or an absence of detected motion. Figure 1 shows an example of estimated bounding boxes for tracked objects and their "heads". The system successfully

[4] Currently, p is set to 10. This is largely dependent on the frame rate of the video signals.

tracks multiple moving objects in the absence of occluding motions. It has been implemented using a Datacube MaxVideo250 and currently tracks at 5Hz.

Tracking faces based upon motion information alone is often insufficient and computationally under-constrained, especially with multiple objects moving closely or under occlusions. If it is assumed that a tracked object is a person, the location of the head can be estimated using some simple heuristics. However, such a crude approach often fails under realistic operational conditions. An additional "face model" has to be present in order to constrain the problem.

3 Model-based face detection

Face detection methods should exhibit invariance to lighting conditions, spatial scale, head pose, small image-plane rotations, hair-style, facial expression and the presence or absence of spectacles and make-up. We give a brief review of methods employed in *static* scenes before describing the method used by our system which is designed to find faces in *dynamic* scenes.

Faces have been detected using simple shape models [18, 29] and symmetry [18]. Colour and texture provide useful cues [11, 30, 38]. Detection of local facial features using photometric measurements seems unreliable and must be coupled with a model of the spatial arrangement of these features [6, 17]. At low spatial resolution such an approach is even less robust.

"Holistic" photometric representations which model the internal structure of faces are more promising for our purposes. A naive example of this approach could be the use of template matching of raw image intensities. Turk and Pentland [35] used "eigenfaces" for detection and in a refinement, probability density estimates for a "face class" were obtained using a principal sub-space of face image space [22]. Burel and Carel [5] used a vector quantization network to cluster face and non-face data while Sung and Poggio [33] used 12 Gaussian clusters (6 face and 6 near-face) to model the face class distribution. In order to obtain a low false-positive rate[5], non-face patterns lying near the true decision boundary were needed. A classifier trained to cope with difficult near-face patterns should also correctly classify easier non-face patterns. Near-face patterns can be selected using an iterative training method in which patterns incorrectly classified as faces are included in future training sets [5, 21, 28, 33]. A weakness inherent in these methods is the representation of images as raster vectors without coding for 2D topology or local spatial arrangement. Neural nets with weight-sharing receptive fields (RF's) have been used to extract translation invariant features [32, 36]. Rowley *et al.* [28] trained nets with square and elongated RF's without weight-sharing and obtained good results.

Our system uses a similar approach to that of Rowley *et al.* [28]. A net was trained to detect frontal and near-frontal views[6]. Face training images were normalised with respect to orientation and scale and the resulting 20×20 windows were masked (Figure 2).

[5] The false-positive rate is the fraction of non-face patterns incorrectly classified as faces.

[6] Other networks could be similarly trained to detect faces at different pose angles in order to build a hierarchical view-based detector exhibiting pose invariance.

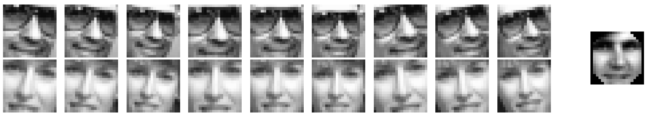

Figure 2: Examples of faces used for training and a masked image.

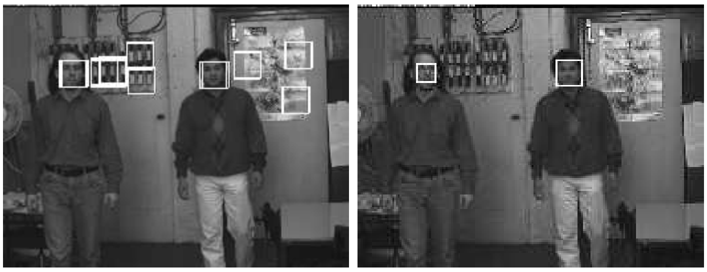

Figure 3: Left: Example output from a static face detector at a given scale. Erroneous detections occur due to "face-like" patterns. Right: Output when the motion-based tracker and the model-based face detector are combined.

One thousand faces were assembled from various face databases[7], the majority from the Usenix database. Non-faces were extracted from 70 images of indoor and outdoor scenes. The training set was expanded (to 9000 images) by rotating faces through 10° and scaling them to 90% and 110%. This forced the net to learn tolerance to some scaling and rotation. Larger scale changes were handled by scanning a pyramid. A multi-layer perceptron with receptive fields was trained using conjugate-gradient descent with iterative selection of near-face patterns.

The left image in Figure 3 shows output at one particular scale and illustrates a weakness inherent in this form of static detector. Since an image contains very many net-sized patches, an extremely low false-positive rate is needed. This rate can be reduced by discarding isolated detections and merging overlapping ones [28]. However, there will always exist non-face patches which when taken out of spatial and temporal context appear "face-like".

4 Tracking faces

In our system, tracking focuses attention for matching by providing scale and location estimates[8] whilst matching provides feedback to the tracking process. Matching need not localise a face correctly in every frame in order for it to be tracked successfully. Face matches can be used to stabilise the tracker. The matching process can also resolve ambiguities in motion grouping. For example, a cluster which is consistently found to possess two heads should probably be split into two clusters. Figure 4 shows some example output.

[7] This work utilised image databases from Olivetti, Usenix and the Universities of Essex, Manchester, Bern and Bochum. We thank those who provided data including Rolf Würtz,

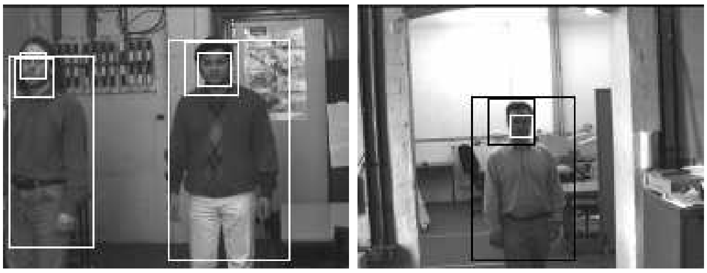

Figure 4: Example output from our system with bounding boxes for objects, heads and faces.

5 GWT face representation and face pose

A Gabor wavelet transform (GWT) yields representations which are locally normalised in intensity and decomposed in orientation and spatial frequency. It thus provides invariance to illumination, skin tone and hair colour as well as selectivity in scale via a pyramid representation. Furthermore, it permits investigation into the role of locally oriented features with regard to pose changes. A GWT is performed by convolution with a set of Gabor functions [12, 39]. Figure 5 shows a face convolved with Gabor kernels at 3 frequencies and 4 orientations varying by 45°. Such a transformation can be regarded as making approximated orthogonal projections onto a set of functions with scale and orientation selectivity. The kernels have zero DC-response and therefore provide invariance to local intensity levels. The real and imaginary parts of the kernel responses tend to oscillate with their characteristic frequency making them highly sensitive to image-plane translations. This property is avoided by taking the magnitude of the responses thereby removing phase information [37, 39] (the lower right row in Figure 5). All our experiments used magnitude responses at a single frequency.

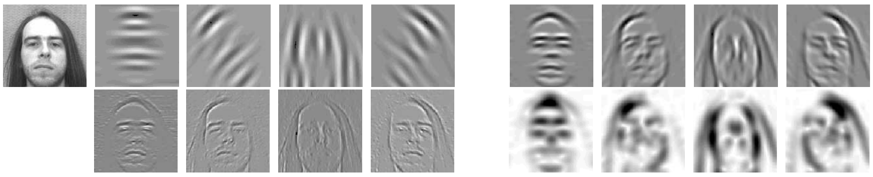

Figure 5: Top left: 4 low frequency responses. Top right and bottom left: responses at higher frequencies. Bottom right: GWT magnitude responses at the middle frequency.

5.1 Face pose eigenspaces

Given an n-frame sequence S=[$S_0, S_1, \ldots, S_{n-1}$] of a head rotating in depth, a Pose Eigen-Space (PES) is calculated by applying PCA to the n frames. Projection of

Henry Rowley, Woodward Yang, Libor Spacek, Andreas Lanitis and Bernard Achermann

[8] Face pose could also be estimated using prediction since it varies smoothly over time.

each frame onto the first few eigenvectors yields a low-dimensional "pattern vector" representation. Projection onto the first three eigenvectors permits visualisation of the distribution of poses in the representation space (see Figure 6).

It is perhaps inappropriate to perform PCA on representations which are not invariant to changes in viewing conditions. We applied PCA to (1) intensity faces, I, normalised by subtracting the mean intensity and dividing by its standard deviation[9] and (2) composite GWT faces $G(I)$ of equal dimensionality to I formed by concatenating four "oriented" GWT faces, each a sub-sampled Gabor response to a different orientation. A principal component (PC) of $G(I)$ can be visualised as a composite "eigen-image" in which a pixel's magnitude is a measure of response variability of a Gabor kernel at the corresponding position in the original image. The magnitudes of the first eigen-image indicate *where* in the image-plane *which* orientations encode the most information about pose.

5.2 Experiments

Several 60-frame profile-to-profile sequences of heads rotating under different lighting conditions were obtained from our head tracker (see Figure 6). In addition, labelled sequences of 12 people were captured in which subjects looked at markers in 10° increments. These images were cropped manually and illumination varied. All images were sub-sampled with smoothing to 64×64 pixels and aligned around the visual centroid of the head. Note that head rotation results in translation of the face.

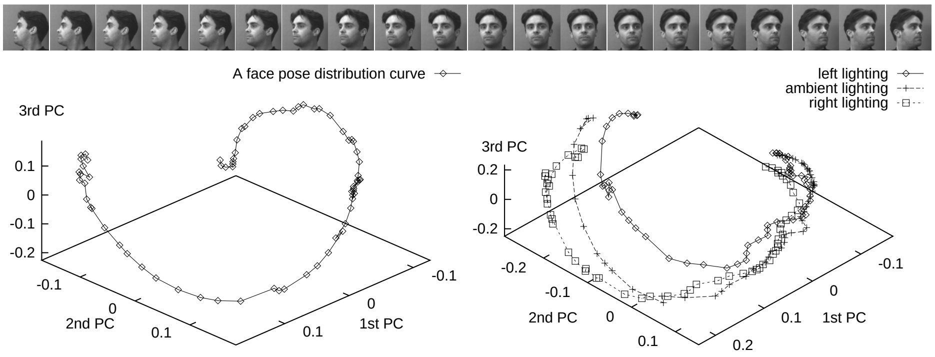

Figure 6: Top: 20 frames from a profile-to-profile sequence. Left: PES of that sequence. Right: manifold formed by 3 faces rotating under different lighting directions

Initially, we consider normalised intensity images. Three unlabelled sequences of the same person under different lighting conditions were projected onto a PES derived from only one of these sequences. Figure 6 shows the resulting curves. They form a fairly smooth manifold parameterised by pose and illumination. The 3rd PC seems to capture lighting changes. This is similar to manifolds obtained in [23] for various non-face objects under robotically manipulated conditions. In

[9] This corrected variations in overall intensity, camera gain and aperture. This is an approximation since factors such as skin tone and hair colour also influence the moments of intensity.

contrast, the face sequences used here were produced by a tracker with left, right and ambient lighting. The resulting manifold is less smooth, reflecting more realistic conditions. A straightforward way to derive a generic PES is to use a mean sequence $\bar{S} = (\bar{I}_0, \bar{I}_1, \ldots, \bar{I}_{n-1})$ containing the mean image at each pose angle over several sequences. The left-hand plot in Figure 7 shows the pose distribution of a mean sequence formed using 11 face sequences. Also plotted are projections into this mean PES of a novel face and a non-face (a fan) rotating similarly from profile-to-profile. The non-face object is distant from the faces for most pose angles. While the 1st PC separates left and right poses, the 2nd and 3rd PCs jointly discriminate between poses from profile to frontal views. This can also be observed from the eigen-images shown above this plot. It is clear that the 4th and 5th PCs capture finer changes in pose angles.We performed PCA similarly with the composite GWT representation. The pictures on the top right of Figure 7 show the first 5 PC's of the mean GWT sequence rotating from $-90°$ to $+90°$. While the horizontal component of the 1st PC plays an important role in dividing the pose angles into two groups, the vertical sub-image (the third in the column) of the 1st PC has relatively little significance. However, vertical orientation becomes a dominant factor in separating pose angles in all other PC's. This is due to the fact

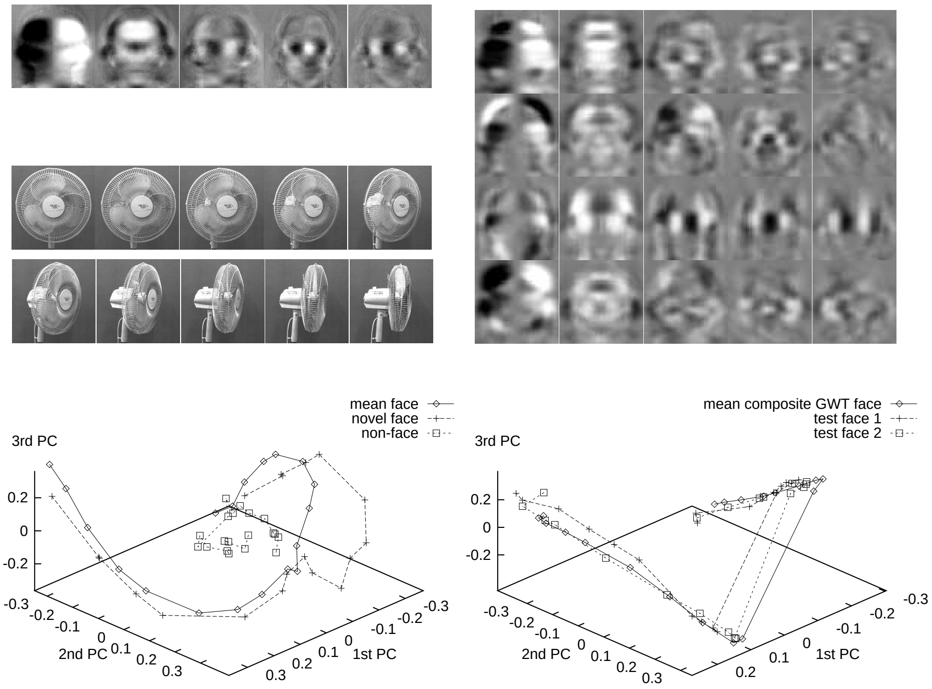

Figure 7: (1) Top left: first 5 PC's of mean faces. (2) Top right: first 5 PC's of the mean GWT faces. Each composite GWT face is a column of responses at 0° (horiz.), 45°, 90° (vert.) and 135°. (3) Plot on left: projections onto first 3 PC's of the mean face sequence, a novel face and the fan shown. (4) Plot on right: Projections of mean GWT face sequence and 2 test face sequences into mean GWT PES.

that the sequences are based on rotation from profile-to-profile. It suggests that in sequences containing pose changes arising from diagonal rotations, the sub-images of PCs that correspond to 45° and 135° orientations (the 2nd and 4th rows) may become more significant. Figure 7 also shows curves in the PES of the mean GWT faces. This plot reinforces our observations regarding the eigen-images. Compared to PES of mean intensity, the curves are well linearised. Pose angles are clearly divided into two groups by the frontal view and are symmetrically distributed along two lines, clearly separable and easily measurable.

6 Conclusions

A real-time multi-motion tracker was implemented using Kalman filters to track objects as groups of temporal zero-crossings. This was combined with a model-based neural net face detector. Motion-based and model-based representations provide complementary approaches to tracking which when combined as a closed-loop system yield more robust solutions than either in isolation. Such a system was described for surveillance. Future work will be concerned with closer integration of the approaches as well as improving their performances in isolation. Models for other face views would increase robustness by providing matches in a higher proportion of frames. Better mechanisms for feedback from the matching to the Kalman filters and the grouping processes also merit investigation.

We introduced a face representation scheme based on a GWT to normalise intensity and scale and to investigate the role of locally oriented features in regularising pose distributions. Pose eigenspaces based on PCA were used to represent and interpret sequences of faces rotating in depth. Whilst the first PC divides all poses from profile-to-profile into two symmetric parts centred at the frontal view, remaining PC's differentiate poses between profile and frontal views. The third PC also seems to capture changes in illumination. The GWT representation gives a highly linear pose distribution. It appears that Gabor kernels of different orientation play some role in "regularising" pose distributions. This is computationally very attractive for determining poses of novel faces. With further study, such a representation could be used to construct a simple generic face pose eigenspace which in turn can be used to estimate poses of unknown faces. This can be done by determining their positions along the pose manifold [23]. Alternatively, the manifold could be modelled probabilistically using mixture density estimation and regression techniques to estimate pose.

In this paper, pose estimation has been treated essentially as a pattern recognition task. There clearly exist, however, contextual cues such as body pose and continuity of pose change which could be used. This will be one of the main focuses of our future work.

References

[1] D. J. Beymer. AI Memo 1461, MIT, 1993.

[2] V. Bruce, A.M. Coombes, and R. Richards. *Image and Vision Computing*, 11, 1993.

[3] R. Brunelli and T. Poggio. *IEEE PAMI*, 15(10), October 1993.

[4] H. Bülthoff, S. Edelman, and M. Tarr. AI Memo 1479, MIT, 1994.

[5] G. Burel and D. Carel. In *Pattern Recog. Letters*, volume 15, pages 963–967, 1994.

[6] M.C. Burl, T.K. Leung, and P. Perona. In *IWAFGR*, Zurich, June 1995.

[7] B. F. Buxton and H. Buxton. *Proc. Royal Society of London*, B-218, 1983.

[8] H. Buxton and S. Gong. *Artificial Intelligence*, 78:431–459, 1995.

[9] N. Costen, I. Craw, and S. Akamatsu. In *ECCV*, Cambridge, England, April 1996.

[10] I. Craw, E. Ellis, and J.R. Lishman. *Pattern Recognition Letters*, 5, Febuary 1987.

[11] Y. Dai and Y. Nakano. In *IWAFGR*, pages 238–242, 1995.

[12] J. Daugman. *J. Opt. Soc. Am.*, 2, 1985.

[13] J. H. Duncan and T.-C. Chou. *IEEE PAMI*, 14(3), 1992.

[14] S. Gil, R. Milanese, and T. Pun. In *ECCV*, volume II, pages 307–320, 1996.

[15] S. Gong and H. Buxton. In *BMVC*, 1993.

[16] S. Gong, A. Psarrou, I. Katsoulis, and P. Palavouzis. In *European Workshop on Combined Real and Synthetic Image Processing for Broadcast and Video Production*, Hamburg, November 1994.

[17] H. P. Graf, T. Chen, E. Petajan, and E. Cosatto. In *IWAFGR*, pages 41–46, 1995.

[18] A. Jacquin and A. Eleftheriadis. In *IWAFGR*, pages 142–147, 1995.

[19] M. Kirby and L. Sirovich. *IEEE PAMI*, 12(1), 1990.

[20] S. McKenna, S. Gong, and H. Liddell. In *2nd Eur. Workshop on Parallel Modelling of Neural Operators*, Nov. 1995.

[21] S. J. McKenna, I. W. Ricketts, A. Y. Cairns, and K. A. Hussein. In *Neural computing - research and applications II.* IOP, 1993.

[22] B. Moghaddam and A. Pentland. In *ICCV*, Cambridge, Massachusetts, June 1995.

[23] H. Murase and S. K. Nayar. *IJCV*, 14, 1995.

[24] A. Pentland, B. Moghaddam, and T. Starner. In *CVPR*, 1994.

[25] C. Ponticos. In *BMVC*, pages 449–458, 1993.

[26] A. Psarrou, S. Gong, and H. Buxton. In *ICNN*, Australia, 1995.

[27] R. P. N. Rao and D. H. Ballard. In *IJCAI*, Montreal, 1995.

[28] H. A. Rowley, S. Baluja, and T. Kanade. Technical Report CMU-CS-95-158R, 1995.

[29] A. Samal and P. A. Iyengar. *Int. J. of Patt. Recog. and A. I.*, 9(6):845–867, 1995.

[30] B. Schiele and A. Waibel. In *IWAFGR*, 1995.

[31] S. M. Smith and J. M. Brady. *Eng. applications A. I.*, 7(2):191–204, 1994.

[32] F. Soulie, E. Viennet, and B. Lamy. *Int. J. Patt. Recog. A. I.*, 1994.

[33] K. Sung and T. Poggio. Technical Report AI Memo 1512, CBCL 103, MIT, 1995.

[34] P. Torr, T. Wong, D. Murray, and A. Zisserman. In *BMVC*, 1991.

[35] M. Turk and A. Pentland. *J. Cog. Neur.*, 3, 1991.

[36] R. Vaillant, C. Monrocq, and Y. Le Cun. *IEE Proc. Vis. Image Sig. Proc.*, 141(4):245–250, 1994.

[37] L. Wiskott, J.-M. Fellous, N. Kruger, and C. von der Malsburg. In *IWAFGR*, 1995.

[38] H. Wu, Q. Chen, and M. Yachida. In *IWAFGR*, pages 314–319, Zurich, June 1995.

[39] R. P. Würtz. *Multilayer Dynamic Link Networks for Establishing Image Point Correspondences and Visual Object Recognition.* Verlag Harri Deutsch, 1994.