View Alignment with Dynamically Updated Affine Tracking

Fernando de la Torre †, Shaogang Gong ‡ and Stephen McKenna ‡

† *Department of Signal Theory, Universitat Ramon Llull, Spain*
‡ *Department of Computer Science, Queen Mary and Westfield College, England*
E-mail: ftorre@salleURL.edu, {sgg,stephen}@dcs.qmw.ac.uk

Abstract

We propose a framework for fast view alignment using adaptive affine tracking. We address the issue of modelling both shape and texture information in eigenspace for view alignment. We present an effective bootstrapping process based on colour segmentation and selective attention. We recover affine parameters with dynamic updates to the eigenspace using most recent history and perform predictions in parameter space. Experimental results are given to illustrate our approach.

1 Introduction

A view-based representation assumes that a piecewise linear vector space exists in which each view is represented by a vector [1]. For object recognition in dynamic scenes using view-based representation, frame to frame view alignment is essential. This requires establishing image correspondences in successive frames of a moving object which may undergo both affine and viewpoint transformations [2]. However, to obtain consistent dense image correspondence is both problematic and expensive since changes in viewpoint could result in self-occlusions which prohibit complete sets of image correspondences from being established. Alternatively, sparse correspondence can be established for a carefully chosen set of feature points [3]. A different approach that is computationally less expensive and does not rely on reliable and fast feature detection in every frame uses holistic appearance-based templates. This assumes that all points of interest of an object move coherently in space and such an approximate rigidity assumption permits a relatively simplistic parametric model to be used for alignment. If the model is also built based on data from a large set of viewpoints, it can in theory recover pose change as well. A good example of this approach is affine tracking in eigenspace, known as EigenTracking [4]. EigenTracking attempted to establish appearance-based correspondence of a moving rigid object by recovering a parameterised affine transform in eigenspace, constructed from object images of different views. However, due to its rigidity assumption and the use of appearance-based templates, in general EigenTracking fails to capture changes which are not sufficiently affine, in particular it copes poorly with objects of irregular shape such as human faces. Furthermore, to be able to establish image correspondence across different viewpoints, EigenTracking requires a training image set for a large number of views. This is impractical and computationally expensive. We propose an integrated scheme for view alignment which (1) uses both shape and texture in eigenspace in a simple manner which could relax the rigidity assumption without introducing too much computational cost, (2) enables effective bootstrapping, (3) estimates parameters with selective attention in a dynamically updated, viewpoint centred eigenspace, and (4) performs parameter prediction.

2 Encoding Appearance and Shape

A set of p images with N pixels given by a matrix $\mathbf{A}$ can be represented by the eigenspace of their covariance matrix $\mathbf{C}$ where usually $p < N$. The image matrix $\mathbf{A}$ can be decomposed using Singular Value Decomposition (SVD) which gives $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathsf{T}}$. The principal subspace comprising the k most significant eigenvectors is used for view alignment. Therefore, in the remainder of this paper, $\mathbf{U}$ is an $N \times k$ eigenvector matrix, $\mathbf{\Sigma}$ is a $k \times k$ diagonal matrix of eigenvalues and $\mathbf{V}$ is a $p \times k$ matrix. An image[1] $\mathbf{I}$ is represented by projection onto eigenvectors $\mathbf{u}_j$, i.e. $\mathbf{I} \approx \sum_{j=1}^{k} c_j \mathbf{u}_j = \mathbf{U}\,\mathbf{c}$ where $k < p$ is the number of eigenvectors actually used and $\mathbf{c}$ gives projection coefficients.

2.1 Appearance-based Affine Tracking

If image changes are approximately affine, correspondence for alignment can be achieved by treating an image

[1] Throughout this paper, we use $\mathbf{I}$ to represent an image vector rather than the Identity matrix. Furthermore, a pre-filtering process is often necessary if global illumination is unstable [3].

$\mathbf{I}(\mathbf{f}(\mathbf{x},\mathbf{a})) = [I(\mathbf{f}(\mathbf{x}_1,\mathbf{a})), I(\mathbf{f}(\mathbf{x}_2,\mathbf{a}))\ldots I(\mathbf{f}(\mathbf{x}_N,\mathbf{a}))]^{\mathrm{T}}$ as a function of an affine transform given by parameters $\mathbf{a} = (a_o, a_1, a_2, a_3, a_4, a_5)^{\mathrm{T}}$, where

$$\mathbf{f}(\mathbf{x},\mathbf{a}) = \begin{bmatrix} a_0 \\ a_3 \end{bmatrix} + \begin{bmatrix} a_1 & a_2 \\ a_4 & a_5 \end{bmatrix} \begin{bmatrix} x - x_c \\ y - y_c \end{bmatrix} \qquad (1)$$

$\mathbf{x}_c = (x_c, y_c)^{\mathrm{T}}$ is the centre position of the object template. Alignment can then be accomplished by recovering both the affine parameters $\mathbf{a}$ and the projection coefficients $\mathbf{c}$ by minimising a cost function $\min_{\mathbf{c},\mathbf{a}} \rho\left[\left(\,\mathbf{I}(\mathbf{f}(\mathbf{x},\mathbf{a})) - \mathbf{U}\,\mathbf{c}\,\right), \sigma\right]$, where ρ is a robust error norm and σ is a scale factor that controls the convexity of the norm [4, 5]. In our scheme, we use the Geman-McClure error norm [6] given by $\rho(x,\sigma) = x^2/(\sigma^2 + x^2)$. Outliers will be considered values from the inflexion point of the norm, which are residuals with $x_i > \sigma/\sqrt{3}$. In general, the above cost function is non-convex and minimisation can result in local minima. A minimisation algorithm found to be effective in this case is gradient descent with the continuation method of graduated non-convexity [6]. It begins with a large value of σ where all the points are inliers. Then σ is successively lowered, reducing the influence of outliers. While it is not guaranteed to converge to a global minimum, the method is effective for visual tracking since continuity of motion provides good starting points.

A dramatic reduction in computational cost is achieved by avoiding image warping in every iteration. This is done by adopting the following linear approximation to the above cost function:

$$\min_{\mathbf{c},\mathbf{a}} \rho\left[\left(\,\nabla\mathbf{I}^{\mathrm{T}}\,\mathbf{f}(\mathbf{a}) + (\mathbf{I} - \mathbf{U}\mathbf{c})\,\right), \sigma\right] \qquad (2)$$

where $\nabla\mathbf{I}$ is the image gradient $[I_x,\, I_y]^{\mathrm{T}}$ [4].

2.2 Encoding Shape Using Landmarks

The difficulty in encoding shape is to be able to compute correspondences quickly and sufficiently robustly. To achieve such a purpose, we propose to learn the coordinates of known landmarks through training images[2]. Let $\mathbf{A} = [\mathbf{I}_1\,\mathbf{I}_2\,\ldots\,\mathbf{I}_p]$ be the matrix with columns of training images and let $\mathbf{X} = [\mathbf{x}_1\,\mathbf{x}_2\,\ldots\,\mathbf{x}_p]$ be the coordinates of the landmarks in these images. The landmarks are assumed to have been located by hand and are the positions of facial feature points. We first took the approach to construct a concatenated matrix $\mathbf{A}^* = [\{\mathbf{I}_1, \mathbf{x}_1\}\,\{\mathbf{I}_2, \mathbf{x}_2\}\,\ldots\,\{\mathbf{I}_p, \mathbf{x}_p\}]$. The new matrix $\mathbf{A}^*$ is a modification of $\mathbf{A}$ with additional feature vectors concatenated to the tail of each training image vector. However, the large scale difference in variances of $\mathbf{A}$ and $\mathbf{X}$ causes numerical problems. To obtain comparable variance, we scale each shape vector $\mathbf{x}_i$ by the maximum norm. A more considered approach to achieve comparable variance between the appearance (image) and shape vectors in eigenspace can be adopted [7]. It is worth pointing out though that better results were actually achieved with modelling the appearance and shape vectors in independent eigenspaces rather than with the concatenated eigenspace.

Once the landmarks have been learned from the training set, we can recall the landmarks during the tracking process. If a new image is aligned with the eigenspace the reconstruction from the coefficients in the eigenspace is given by $\mathbf{U}\mathbf{c}$. To recall the most likely landmark positions for the new image based on what has been learned in training, an inverse transformation between the eigenspace and the training set (related by the SVD) is performed on the shape components only:

$$\mathbf{x}_{new} = \mathbf{X}\,(\mathbf{V}\Sigma^{-1}\mathbf{c}) \qquad (3)$$

$\mathbf{V}\Sigma^{-1}\mathbf{c}$ are the p coefficients which best reconstruct the new image in a least-squares sense from the training data, and $\mathbf{V}\Sigma^{-1}$ can be pre-computed off-line in order to speed up the tracking process. Given sufficient training examples with known landmarks, incorporating shape in the eigenspace could enable previously learnt feature positions to be recalled during tracking, therefore avoiding the need to perform online feature detection and correspondence which are computationally both expensive and problematic.

3 Bootstrapping

Colour-based segmentation can provide robust and very fast focus-of-attention for the initialization of the affine parameters [8]. Here we adopt multi-colour Gaussian mixture models to perform real-time object detection and focus of attention. The mixture models were estimated in two-dimensional hue-saturation colour space. Such representations are chosen to permit some level of robustness against brightness change. Probabilities are computed for pixels in an image search space and the size and position of the object are estimated from the resulting probability distribution in the image plane [8]. An example of colour-based, real-time, coarse segmentation using a mixture of four Gaussians can be seen in Figure 1.

3.1 Adaptive Attentional Window

The most computationally expensive operation in recovering affine parameters is to recursively warp the image relative to its center in order to minimise the cost function of Equation (2). To address this problem, the affine transformation is only computed within an attentional window. The size of this window adapts to the size of the object. Affine transforms are performed relative to the center of the window which must coincide with the centroid of the object in order to minimize the errors in estimated rotation and scale

[2] The same training images are used for constructing the eigenspace.

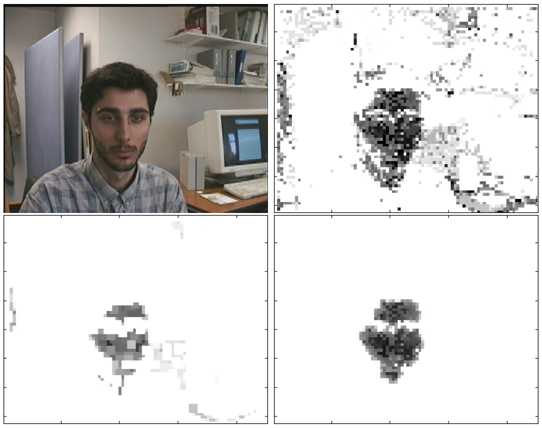

Figure 1. Top row: An image and object foreground probabilities in the image plane. Bottom row: results of segmentation with multi-resolution relaxation after 1 and 4 iterations.

parameters. The centre of the attentional window is estimated using prediction (see Section 5).

3.2 Morphological operations

Colour provides only a crude initial estimate for an attentional window. An improved estimate is obtained in a computationally efficient way by applying recursive nonlinear morphological operations at multiple resolutions. The method can be seen as a combination of relaxation and something similar in spirit to geodesic reconstruction in morphology [9]. This "geodesic relaxation" algorithm is as follows:

(a) Compute log probabilities in a 1/4 sub-sampled image and normalise these probabilities to give a low resolution grey-scale "probability image" $\mathbf{I}^o$. (b) Apply grey-level morphological erosion to $\mathbf{I}^o$. This reduces noise and erroneous foreground and yields an image $\mathbf{I}^{er}$. (c) Let $\mathbf{I}^* = \mathbf{I}^{er}$, then apply the following operation a fixed number of times:

$\mathbf{I}^* = \frac{1}{2}(\mathbf{I}^* \otimes \texttt{low-pass-filter} + \mathbf{I}^o)$

where $\otimes$ denotes convolution.

The resulting image $\mathbf{I}^*$ (see Figure 1) is used to fit a bounding box which is then used as an initial attentional window. The iterative process is fast because good results are obtained in a few iterations using low resolution images.

To give an initial estimation of the affine scale parameters (a_1, a_5), morphology and colour are used to estimate the eye-mouth region within the attentional window (see Figure 2). The process is as follows:

(a) Perform vertical erosion on the 1/2 sub-sampled and thresholded attentional window $\mathbf{I}^{win}$ to give $\mathbf{I}^{er}$. (b) Perform geodesic reconstruction of $\mathbf{I}^{er}$ with $\mathbf{I}^{win}$ as the reference image to give $\mathbf{I}^{rec}$. (c) Compute $\mathbf{I}^{end} = \texttt{opening}(\mathbf{I}^{win} - \mathbf{I}^{rec})$

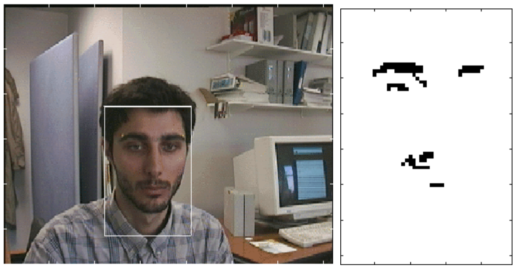

Figure 2. Left: The attentional window estimated using colour cues. Right: The main facial feature region extracted using morphological operators.

3.3 Parameter Initialisation

Colour and morphological operators provide an initial attentional window and approximate estimates of translational and scale parameters. These initial parameter estimates are further refined by recursively applying Successive-Over-Relaxation [6] to the same initial image in order to minimize the cost function (2). Robust norms with a continuation method and a multi-resolution representation were used in order to avoid local minima. At first, only the translational parameters (a_0, a_3) were optimised in order to align the centre of the attentional window with the eigenspace. Subsequently, the scale parameters (a_1, a_5) were optimised. It was assumed that affine rotation was negligible in the initial frame. An example of this parameter initialisation process is shown in Figure 3. In this example, the initial estimates provided by the colour model were unusually poor.

4 Adaptive Pose-Driven Affine Tracking

The EigenTracking method made use of a fixed, global eigenspace (GES) representation for reconstruction and tracking. This eigenspace was built by performing SVD on a relatively large, fixed training set. The alternative method described in this section yields faster and more accurate reconstruction. It involves the use of local eigenspace (LES) representations built using subsets of the original training set. In particular, at each time frame t, the q training images which are "closest" to the previous affine-normalised, tracked image $\mathbf{I}_{t-1}$, are selected. A new LES is then computed from these q images. The image $\mathbf{I}_{t-1}$ can also be included in the set used to compute the LES and this helps to compensate for temporary changes not represented in the original training set (e.g. unusual facial expressions).

The computation of a new LES in (potentially) every frame might seem prohibitively expensive. However, the iterative matching algorithm typically converges more quickly when reconstructions are performed using an LES. In practise, this faster convergence more than compensates for the expense of computing the LES. The overall result is

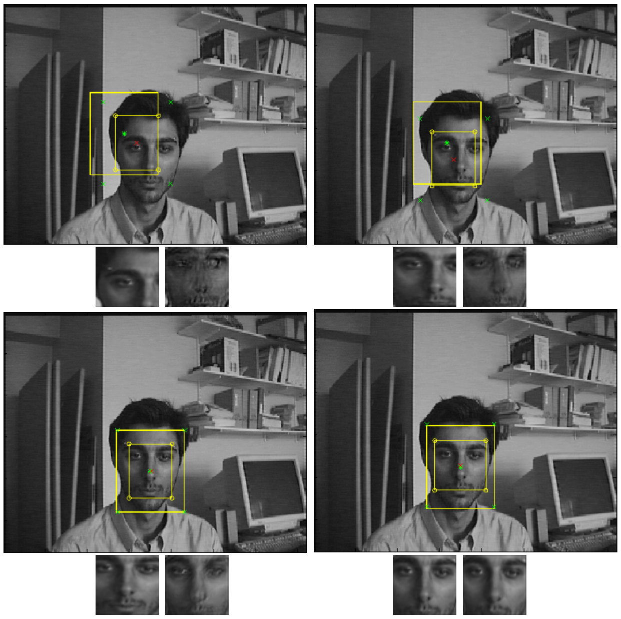

Figure 3. Affine parameter initialisation: The attentional window is overlaid with a smaller box indicating estimated translation and scale parameters. Below each frame is the located region (left) and its reconstruction (right).

faster and gives more robust tracking. The q selected training images are usually images with similar 3D pose and facial expression and can therefore be accurately represented using only a few eigenvectors. In order to achieve good enough reconstruction, enough eigenvectors are retained to account for 95% of the variance in the training set.

In order to compute an LES, the q "closest" training images must be selected. An obvious way in which to perform this selection is to measure the Euclidean distance between $\mathbf{I}_{t-1}$ and each of the training images and to select the q nearest images. These distance measurements can be efficiently approximated using projections onto the precomputed GES [10]. Further efficiency can be obtained using a multi-resolution scheme in which $\mathbf{I}_{t-1}$ is sub-sampled and projected onto a precomputed GES of equally low resolution.

If an ordering can be imposed on the training set, however, an alternative scheme becomes possible. The closest match in the training set is then used to index into this ordered set and the q images for the LES are selected using the predetermined ordering[3]. For example, if the training set consists of a sequence of a head rotating from left to right then time imposes a natural ordering. The nearest match then yields an estimation of head pose and the LES is computed from images of similar pose.

Another way to derive find the closest matches uses the information of the projection coefficients and their relationship with the training set. The p coefficients $\mathbf{y} = [y_1\, y_2\, \ldots\, y_p]^{\mathrm{T}}$ whose linear combination of the training set minimises the Euclidean distance, $\min_{\mathbf{y}} \parallel \mathbf{I}_{t-1} - \mathbf{A}\mathbf{y} \parallel^2$, are given by:

$$\mathbf{y} = \mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{c}, \qquad \mathbf{y} = (\mathbf{A}^{\mathrm{T}}\mathbf{A})^{-1}\mathbf{A}^{\mathrm{T}}\mathbf{I} \tag{4}$$

As we are working with normalized images, minimising the Euclidean distance is identical to maximising the dot product. Note that we can obtain the dot products if the coefficients in the eigenspace $\mathbf{c}$ are known, that is:

$$\mathbf{A}^{\mathrm{T}}\mathbf{I} = (\mathbf{A}^{\mathrm{T}}\mathbf{A})\mathbf{y} = \mathbf{V}\mathbf{\Sigma}\mathbf{c} \tag{5}$$

The position of the maximum component of $\mathbf{A}^{\mathrm{T}}\mathbf{I}$ corresponds to the "closest" image in the training set. It can be easily shown that this is mathematically equivalent to [11]. However, our approach establishes correspondence between the eigen-coefficient and the training images directly and is less expensive for computing the GES[4].

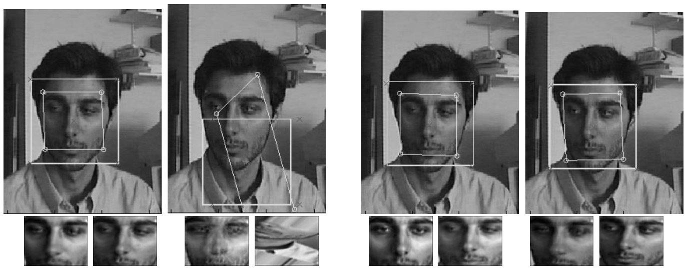

Figure 4. Left two images: EigenTracking without dynamic updates fails due to insufficient training views. Right two images: The adaptive scheme.

5 Prediction of Six Affine Parameters

In order to cope with displacements of more than a few (3-4) pixels between frames, it was necessary to use prediction. In each frame, the affine parameters were predicted and these predictions were used to initialise the iterative optimisation algorithm. The six affine parameters were predicted using a Kalman filter [12]:

$$\mathbf{x}_{k+1} = \mathbf{\Gamma}\mathbf{x}_k + \mathbf{B}\mathbf{n}_k, \quad \mathbf{z}_k = \mathbf{H}\mathbf{x}_k + \mathbf{r}_k \tag{6}$$

where $\mathbf{x}$ and $\mathbf{z}$ were 12-dimensional state and measurement vectors and $\mathbf{n}_k$ and $\mathbf{r}_k$ were zero-mean white noise with covariance matrices $\mathbf{Q}_k$ and $\mathbf{R}_k$. The dynamic system model

[3] Many strategies have been suggested for performing such a nearest neighbour search. See [11] for a recent discussion.

[4] Computing $\mathbf{C} = \mathbf{c}_1\mathbf{c}_2 ... \mathbf{c}_p = \mathbf{U}^{\mathrm{T}}\mathbf{A}$ requires kpN multiplications. With SVD of A, computing $\mathbf{C} = \mathbf{\Sigma}\mathbf{V}^{\mathrm{T}}$ needs just k^2p operations which is more efficient since $k \ll N$.

used assumed constant velocity [5]. The elements of the vectors $\mathbf{x}$ and $\mathbf{z}$ corresponded to the affine parameters and their velocities.

Appropriate noise covariance matrices were estimated using the EM algorithm with the following least-squares approximations of observation noise and state noise:

$$\mathbf{r}_k \approx \mathbf{z}_k - \mathbf{H}\mathbf{x}_k^-, \quad \mathbf{n}_k \approx (\mathbf{B}^{\mathrm{T}}\mathbf{B})^{-1}\mathbf{B}^{\mathrm{T}}(\mathbf{x}_{k+1}^+ - \mathbf{\Gamma}\mathbf{x}_k^+)$$

where $^+$ denotes *a posteriori* estimation and $^-$ denotes *a priori* estimation. The estimated covariance noise between the affine parameters was also verified visually by plotting pairwise parameter observations. An alternative approach is to derive an optimal estimate based on the Kalman filter's underlying cost function. Estimation of the state vector $\mathbf{x}_{k+1}$ based upon previous measurements is equivalent to minimising cost function:

$$\begin{aligned} E(\mathbf{x}_{k+1}^+) = & \tfrac{1}{2}(\mathbf{x}_{k+1}^+ - \mathbf{x}_{k-1}^-)^{\mathrm{T}}(\mathbf{P^{-1}})(\mathbf{x}_{k+1}^+ - \mathbf{x}_{k-1}^-) \\ & + \tfrac{1}{2}(\mathbf{z}_k - \mathbf{H}\mathbf{x}_{k+1}^+)^{\mathrm{T}}(\mathbf{R^{-1}})(\mathbf{z}_k - \mathbf{H}\mathbf{x}_{k+1}^+) \end{aligned}$$

The first term specifies the temporal constraint while the second expresses the data conservation, where all errors are measured using the Mahalanobis distance. We propose apply one robust norm to the second term in order to derive a robust Kalman filter. Efficient minimization of this function could be performed using a technique such as Iteratively Recursive Least Squares (IRLS) [13].

The human head will inevitably move in ways which are not predictable using these simple dynamic models. An effective way in which to detect this "unpredictability" is to run one iteration of the optimisation algorithm used for tracking. Only if the direction in affine parameter space of this iteration step agrees with that of the Kalman prediction are the predictions utilised. This works well if not many outliers are present because with the continuation method used the initial estimation of the affine parameters with high σ could be quite different from the final one with low σ.

6 Experiments and Discussion

The system was initially implemented in Matlab and took an average of 14 sec/frame. In C, it could run at near 2 sec/frame on a standard 200MHz PC. Applying IRLS to the minimisation process can solve an approximation of the robust formulation in near real-time.

Figure 5 shows the ability of the new adaptive scheme to align a face undergoing non-rigid expression change. Similar problems occur when the assumption of affine transformation is no longer valid. The adaptive scheme was shown to be able to overcome the problem when changes in pose were not sufficiently captured by the training data (see examples in Figure 4).

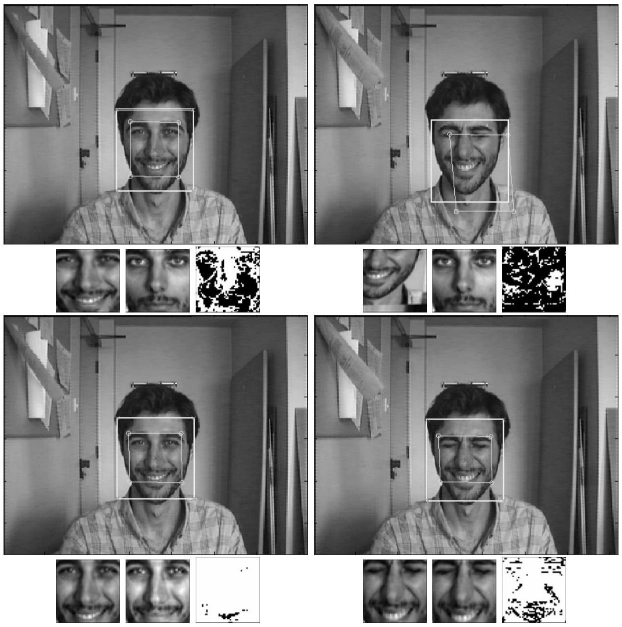

Figure 5. This sequence demonstrates the advantage in using adaptive scheme with non-rigid expression changes. The first row shows the overlaid results from EigenTracking whilst the second row shows the results from the adaptive scheme with shape encoded.

Figure 6 shows view alignment from a 260 frame sequence with both affine and pose variations. The training set had 100 images and 54 eigenvectors were used in a GES capturing 95% of the variance. However, with the adaptive scheme using LES, only 6 eigenvectors were needed to recover sufficiently accurate parameters for alignment.

Figure 7 gives an indication of savings in computational cost when the adaptive scheme is applied. It shows the time taken in seconds to perform the minimisation (Equation (2)). The first column shows the time required for each frame using GES with 54 eigenvectors. The second column shows the time required for each frame with LES but without the previous history at $t-1$. The third column shows the result from augmented LES using $t-1$ tracked data. The mean cost for GES was 29 sec/frame, 15 sec/frame for LES and 12 sec/frame for $t-1$ augmented LES.

In this paper we present an integrated scheme for view alignment. We exploit the transformation between the training set (TS) and the eigenspace in a computationally inexpensive manner in order to establish the correspondence between the landmarks in the TS and the image. A dynamically adaptive scheme was adopted to compensate the

[5] Pairwise plots of the affine parameter measurements typically revealed trajectories in the 6-dimensional affine parameter space which were approximately linear or piecewise linear. Therefore, it seemed reasonable to use a constant velocity model.

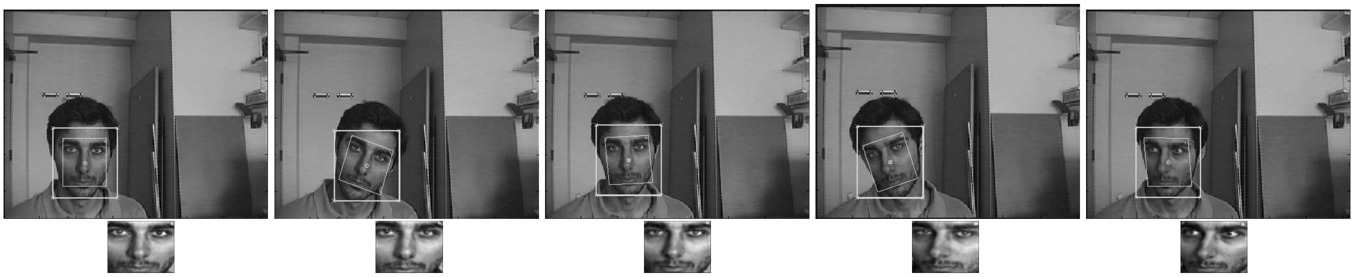

Figure 6. A sequence with attentional windows and face boxes overlaid.

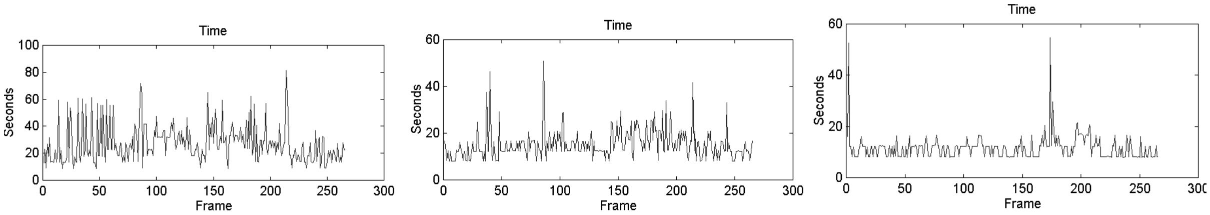

Figure 7. Convergence times required by the minimisation process.

small changes in illumination and the nonlinear transformations that cannot be recovered with global affine transformations. Different approaches were experimented for updating with frame-based updates gave the best results. However, if the training images are ordered, it is possible to update the eigenspace which would be even more desirable [14]. Our current scheme did not include a pre-filtering process to address changes in global illumination. This can be addressed by estimating the illumination in the new image dynamically with the homomorphic filtering and then apply it to the training set. Other way to resolve the problem is to use an additional basis to represent all the possible illumination situations [15]. Gabor wavelets representation can also be employed which gives certain degree of invariance to global illumination change [3].

References

[1] S. Ullman and R. Basri, "Recognition by linear combinations of models," *PAMI*, vol. 13, no. 10, pp. 992–1006, 1991.

[2] D. Beymer and T. Poggio, "Image representations for visual learning," *Science*, vol. 272, pp. 1905–1909, June 1996.

[3] S. McKenna, S. Gong, R. Wurtz, J. Tanner, and D. Banin, "Tracking facial feature points with gabor wavelets and shape models," in *AVBPA*, Switzerland, 1997.

[4] M. Black and Y.Yacoob, "Eigen tracking: Robust matching and tracking of articulated objects using a view-based representation," in *ECCV*, Cambridge, England, April 1996.

[5] P. Huber, *Robust statistics*, John Wiley and Sons, 1981.

[6] M.J. Black and P. Anandan, "The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields," *CVIU*, vol. 63, no. 1, pp. 75–104, 1996.

[7] N. Sumpter, R. Boyle, and R. Tillett, "Modelling collective animal behaviour using extended point distribution models," in *BMVC*, Colchester, September 1997, pp. 242–251.

[8] Y. Raja, S. McKenna, and S. Gong, "Tracking and segmenting people in varying lighting conditions using colour," in *FG '98 (These Proceedings)*, 1998.

[9] L. Vincent., "Morphological grayscale reconstruction in image analysis: Applications and efficient algorithms," *IEEE Trans. on Image Processing*, vol. 2, no. 2, pp. 176–201, 1993.

[10] H. Murase and S. Nayar, "Detection of 3d objects in cluttered scenes using hierarchical eigenspace," *Pattern Recognition Letters*, vol. 18, pp. 375–384, 1997.

[11] S. Nene and S. Nayar, "A simple algorithm for nearest neighbor search in high dimensions," *PAMI*, vol. 19, no. 9, 1997.

[12] S. Kay, *Fundamentals of statistical signal processing: Estimation theory*, Prentice Hall, 1993.

[13] Z. Zhang, "Parameter estimation techniques: A tutorial with application to conic fitting," *IVC*, 1996.

[14] M. Gu and S. Eisenstat, "A stable and fast algorithm for updating the singular value decompositon," Tech. Rep., Yale University, 1994, YALEU/DCS/RR-966.

[15] G. Hager and P. Belhumeur, "Real-time tracking of image region with changes in geometry and illumination," in *CVPR*, 1996.