A Hierarchical Model of Dynamics for Tracking People with a Single Video Camera

I.A.Karaulova[1], P.M.Hall[2], A.D.Marshall[1]

1 Department of Computer Science
University of Cardiff
Cardiff, CF24 3XF, UK
{J.A.Karaulova, dave}@cs.cf.ac.uk

2 Department of Mathematical Sciences
University of Bath
Bath
maspmh@maths.bath.ac.uk

Abstract

We propose a novel hierarchical model of human dynamics for view independent tracking of the human body in monocular video sequences. The model is trained using real data from a collection of people. Kinematics are encoded using Hierarchical Principal Component Analysis, and dynamics are encoded using Hidden Markov Models. The top of the hierarchy contains information about the whole body. The lower levels of the hierarchy contain more detailed information about possible poses of some subpart of the body. When tracking, the lower levels of the hierarchy are shown to improve accuracy. In this article we describe our model and present experiments that show we can recover 3D skeletons from 2D images in a view independent manner, and also track people the system was not trained on.

1 Introduction

This paper introduces a novel model of human dynamics that allows view independent tracking of a 3D human body skeleton in monocular video sequences. The model represents the body dynamics of a collection of people. It is trained on real life examples using Hierarchical Principal Component Analysis (HPCA) to encode geometry and kinematics, and Hidden Markov models (HMM) to encode dynamics. The model can be trained on either 2D or 3D data. It allows us to recover 3D skeletons from 2D image sequences, to track unknown people, and improve tracking accuracy.

Tracking humans in video has applications in many areas including surveillance, computer games, films, and biodynamics. There is a large body of work related to tracking human motion in 3D. We are interested in general methods that allow one to track the whole body, rather than in specialised trackers for face, hands, *etc.* [10]. Encouraging results have been achieved in tracking whole body human motion in 3D using multiple cameras [1, 5, 11]. We are, however, interested in recovering 3D human motion from only one view.

To recover a 3D human skeleton pose on the basis of 2D data we need to know how the 3D human skeleton and 2D data are correlated. Goncalves *et al.* [4] utilised the correlation between the real human arm size and the size of the arm in the image in order to recover its 3D positions. This approach is, however, limited only to a person whose arm geometry was used in the system. In our proposed system this limitation was overcome by embedding into the system the dynamics and geometry of several people and thus making it more general. Bowden *et al.* [2] encapsulated the correlation between 2D image data and 3D skeleton pose in the hybrid 2D-3D model trained on real life examples. The model they used allows 3D inference from 2D data, but their method does not generalise easily to new camera positions, because the 2D part of their model is not invariant to viewpoint.

Another useful feature when tracking objects in video sequences is a model of the object dynamics. Goncalves *et al.* [4] used a Kalman filter for arm tracking, which is a very general mechanism and doesn't describe the way people move. In recent years HMMs have been applied to human behaviour prediction and recognition [14, 15], and are becoming recognised a valuable mechanism for modelling human motion from real life data. We combine the use of HMM with the condensation algorithm [9] to track model states, as do Ong and Gong [12].

Hogg [6] created a view invariant model of a human body. When tracking, it was projected onto the frames in the video sequence to choose the best fitting pose of the model. We act likewise. For Hogg, the space of valid poses was "hard-wired" into the model, rather than learnt from examples and there was no model of the dynamics of a human body. We build on Hogg's work by learning the valid poses from examples, and using HMMs for a description of dynamics.

We describe the structure of our hierarchical model of dynamics in Section 2, look in detail at the tracking process in Section 3, present our 2D and 3D experiments in Section 4 and conclude in Section 5.

2 Hierarchical Model of Human Dynamics

A natural and common way to represent the human body is with connected parts. For example, a lower limb is connected to an upper limb, which in turn connects to the torso. Such models are often used in computer graphics [16]. However, our model is based on "part-of" relationships. For example, a lower limb and an upper limb are part of a whole limb, which in turn is a part of a whole body. We use a "part-of" decomposition because, as we explain below, our model of a collection of people comprises a hierarchy of eigenspaces in which a "high-level" eigenspace contains the major components of a "lower-level" eigenspace. As we explain below, these eigenspaces are used to specify valid *poses* for a collection of people performing a particular action (such as walking or jumping). We regard the transition from one pose to the next as the *dynamics* of the action, and encode this using Hidden Markov Models (HMM). We train our model, both poses and dynamics, from real data. Next we describe the model of valid poses, and then move on to describe the HMMs for dynamics.

2.1 A model of valid poses

It is convenient to begin the description of our model by considering a model of an individual person in a particular pose (as in Figure 1), and use this to develop the content in

the root node of our hierarchy. We mark three-dimensional (3D) vertices, $\mathbf{x} \in \Re^{\mathbf{3}}$ at well defined locations, such as the knee and elbow. Over the whole body there are N such vertices, which we collect into a vector $\mathbf{p} \in \Re^{\mathbf{3N}}$. This vector encodes the geometry of the body. As the individual performs an action the vector varies in time and hence is a continuous function $\mathbf{p}(t) \in \Re^{3N}$. We sample it at M points in time (typically in each frame of a sequence) to obtain a discrete set of poses $\{\mathbf{p}_t\}$. This encodes kinematics.
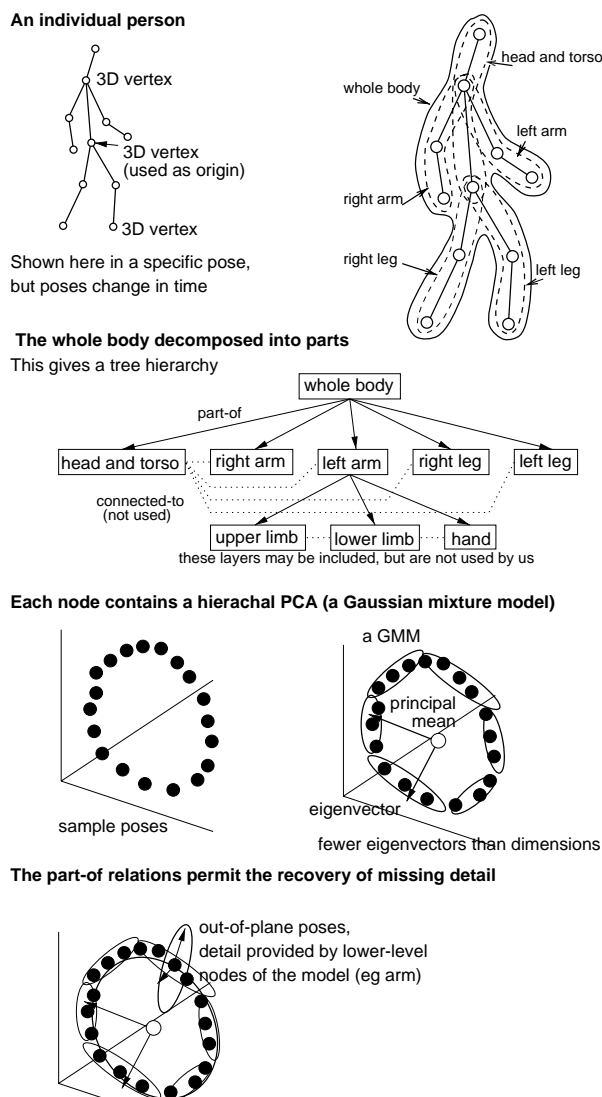

Figure 1: Basic model of a human body

We wish to model the poses (and, later, dynamics) for a collection of K individuals, and so must represent the collection $\{\mathbf{p}_{i,t}\}$, where the subscript i refers to a particular individual. This set samples the distribution of valid skeleton poses and can be represented by a $(3N \times MK)$ matrix, $\mathbf{P}$. It is captured from real data using a variety of vision

systems. (Typically, in our experiments we capture between 200 and 1000 vectors.) This distribution is highly non-linear, due to geometrical and physical constraints on the valid positions of vertices, therefore we model this distribution using Hierarchical Principal Component Analysis (HPCA).

HPCA was originally developed by Heap *et al.* [8], and later utilised by Ong *et al.* [12] for learning the state space of their model. The HPCA method consists of the following steps:

1. Remove dimensions representing small variations in pose by standard PCA, so that the distribution of $\{\mathbf{p}_{i,t}\}$ is represented by the eigenspace model (eigenmodel)

$$(\bar{\mathbf{p}}, \mathbf{U}, \mathbf{\Lambda}, MK)$$

   in which $\bar{\mathbf{p}}$ is the mean of the set, $\mathbf{U}$ is a $(3N \times s)$ matrix of eigenvectors, $\mathbf{\Lambda}$ are the eigenvalues, and MK is the set cardinality; note $s \leq \min(3N, MK)$

2. Projecting the original data set into this eigenspace to acquire dimensionality-reduced samples,
$$\mathbf{r}_{i,t} = \mathbf{U}^T (\mathbf{p}_{i,t} - \bar{\mathbf{p}})$$

3. Cluster the projected data into a number of Gaussian distributions, each represented by its mean and covariance matrix, thus creating a Gaussian Mixture Model (GMM). Each cluster, q_k can also be represented by an eigenmodel

$$q_k = (\bar{\mathbf{r}_k}, \mathbf{V}_k, \mathbf{\Sigma}_k, N_k)$$

PCA's are often used to constrain variations, and hierarchical PCA improves the specificity [8] of this and better models any non-linearities in the system.

Thus far we have considered only the root node of our model. As mentioned, nodes below the root correspond to major body parts, such as the arm, as in Figure 1. The pose of such a part can be represented as a vector, the elements of which come from the whole body vector. Consequently, a collection of poses for a part (the collection ranging over time and individuals) can be treated in exactly the same way as the set of poses for the whole body, that is modelled as a GMM.

Thus our model comprises a hierarchy of nodes, with a hierarchy of eigenspaces in each node (Figure 1). When we refer to "the eigenspace of a node" we mean the root eigenspace in the hierarchy of eigenspaces at that node. The eigenspaces of nodes at lower levels are partially contained within those eigenspaces of nodes at higher levels, thus forming a dependency. Eigenspaces in nodes at the same level are independent (orthogonal). This representation is advantageous: because of dimensionality reduction the eigenspaces in nodes at the higher-levels encode only the major variants of valid poses, the lower-level nodes encode minor variations, and hence capture detail that would otherwise be lost, in a compact way. We make use of this when tracking humans, as explained in Section 3. Overall, our model greatly improves specificity and yet retains the advantages of PCA.

2.2 Modelling Dynamics

In our model GMMs capture the variety of poses the skeleton can have, but we also would like to have a mechanism, which given a skeleton pose at time t would be able to predict

what pose the skeleton is likely to acquire at time $t+1$. For this purpose we adopt Hidden Markov Models (HMM).

HMMs have been used for some time in the speech processing [3] representing possible transitions from one sound into another. Recently they have found use in computer vision for interpreting and predicting human behaviour [14, 15]. Currently, for reasons of simplicity, we use an HMM only in the root node of our hierarchical model. We will comment further on this in our conclusions (Section 5).

A continuous observation HMM consists of the following elements:

- t_1, t_2, t_3, t_4 ..., which are discrete clock times.

- $q_1, q_2, ..., q_N$, which are a number of discrete states. In our case each state is represented by an eigenmodel within a GMM. At each clock time t a new state is possibly entered.

- $\mathbf{A} = \{a_{ij}\}$, $a_{ij} = p(q_j$ at time $t+1|q_i$ at time t), which are the probabilities of transitions between states.

- $\mathbf{B} = \{b_j(o)\}$, where $b_j(o) = p(o_t|q_j$ at time t) is an observation density distribution at state j, which is just the probability that a sample o_t belongs to state j.

- $\mathbf{\Pi} = \{\pi_i\}$ - initial probabilities of being in state i at time $t = 1$.

We initialise the matrix of possible transitions with all elements equal, then we use the Baum-Welch [13] iterative method for estimating the transition probabilities on the basis of the real data.

So far we have described our hierarchical model which represents the geometry and dynamics of a collection of people. In the next section we explain how we use this model for tracking.

3   Tracking Human Skeleton in a Video Sequence

We aim to track a skeleton in image sequences, from frame to frame. In principal the images can be three dimensional (perhaps acquired from a body scanner) or from two dimensional image sequences as obtained from a video camera; our tracking method is largely independent of image modality. This is because we track skeletal poses in the model just described, and use data only to choose between a set of poses that have been generated using the condensation algorithm [9].

Our tracking process can be thought of as a multi-level refinement procedure. It starts by estimating the pose of the whole skeleton using the HMM and the eigenspace at the root node of our hierarchical model. Then it refines the poses of the skeleton parts using the eigenspaces of the corresponding model nodes (Figure 2).

It is convinient to start describing the tracking procedure with describing the estimation of the whole skeleton pose. This step involves using the condensation algorithm in combination with HMM to generate a set of skeleton poses in the top-level node eigenspace. These skeleton poses are then reconstructed to their original space and weighted according to how well they fit the data in the current input frame. The skeleton pose for the current frame is estimated as a weighted mean of the reconstructed set.
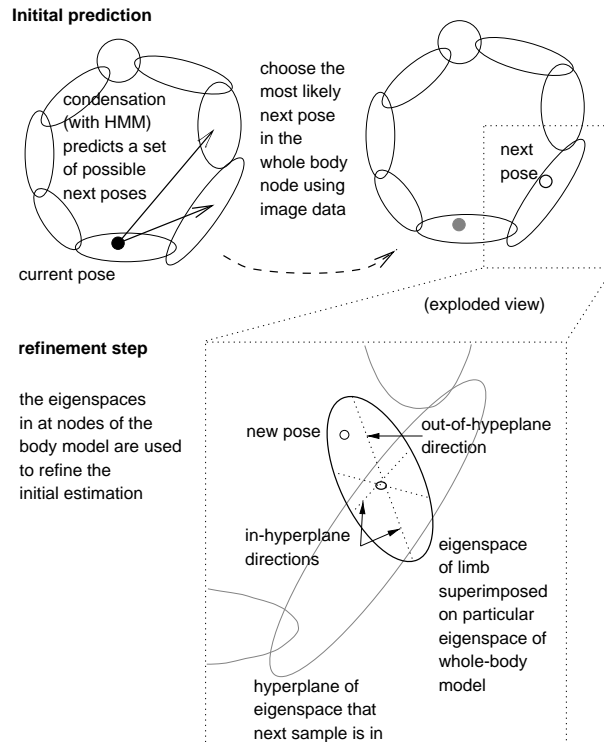
Figure 2: Tracking process with refinement step

Refinement of a particular skeleton part pose is performed in the following way. The estimated pose of this skeleton part is passed from the previous refinement stage. This pose is projected into the eigenspace of the corresponding model node and the probability of it belonging to each cluster of this eigenspace are estimated. A set of samples from each cluster in the eigenspace is then generated, the number of samples belonging to each cluster proportionate to the obtained probabilities. The samples of the skeleton part poses are then reconstructed to their original space and each sample is assigned a weight according to how well it fits the data in the current input frame. The refined skeleton part pose is estimated as a weighted mean of the reconstructed set.

4 Experiments

In this article we restrict ourselves to considering walking motion of a small sample of people. We performed a number of experiments in both 2D and 3D. The experiments in 3D show that we are able to recover 3D configurations of the skeleton on the basis of previously unseen 2D image data, invariant of the camera view. The 2D experiments show that the system is able to track both people it has been trained on and people it has not been trained on. We also showed that using our hierarchy noticeably improves the precision of tracking in 2D. To monitor the precision of tracking we computed the error for each skeleton vertex as the Euclidian distance between the tracked vertex position and

the ground truth vertex position. We experimented with different coordinate systems for representing 3D and 2D skeleton vertex positions, including Cartesian, Polar, and Twist representations, but so far we have found Cartesian coordinates to give the best results in the experiments. We also experimented with including vertex velocities in our data set, but this did not provide significant improvement in tracking.

4.1 3D Experiments

The data consists of 320 frames of a walking 3D human skeleton which was captured using an optical marker-based system consisting of 8 cameras[1]. The human skeleton is represented by 32 vertices and connecting bones (Figure 4). The configuration of the skeleton in each frame is represented by a state vector consiting of 3D Cartesian coordinates of each vertex. The data we used for training is the 3D data from 200 frames, the rest of the frames was used for testing. The 2 sets of 2D testing data were obtained by parallel projection of the rest of the 3D data frames into side and front camera views.

Our hierarchy comprises two levels. The root contains the GMM and HMM for a whole human skeleton. The second level consists of five nodes; one for the right leg, one for the left leg, one for the right arm, one for the left arm, and one for the torso and the head. Each node contains the GMM for the body part.
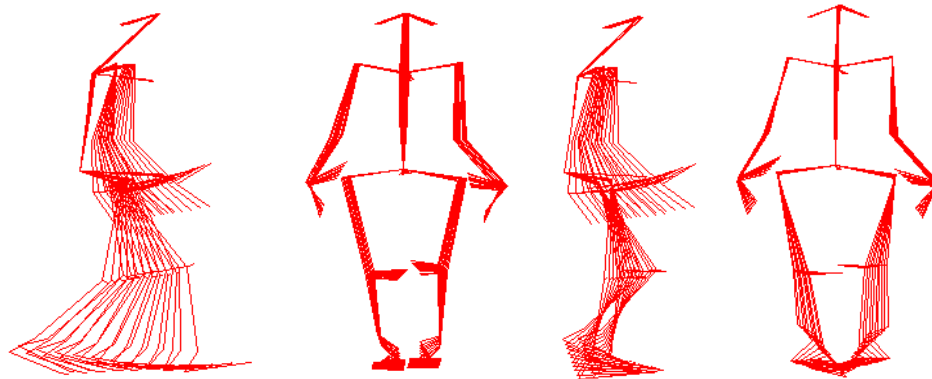

Figure 3: Two main modes of 3D variation in the global eigenspace, side and front views.

We trained our model on 200 frames, keeping 90% of the eigenenergy in the root, leaving just two eigenvectors. The first of these vectors describes forward-backward motion of rigid arms and legs. The second of the vectors describes the degree of bending of the knees and elbows (see Figure 3). We kept 95% of the eigenenergy in the remaining nodes, and used 50 eigenmodels in each of the GMMs. In our experiments the transition probabilities in HMM seem to have settled to reasonable values just after 4-5 iterations.

We tracked the 3D skeleton in both of the projected sequences (side and front views) using only one sequence at a time (Figure 4). When tracking in the side view the average error in the image plane over all vertices and frames was 4 pixels with a spread of about 1 pixel, with the vertical size of the whole figure being 160 pixels. The precision is better for the upper part of the body including arms but worse for the legs. When the recovered 3D model was projected into the front view, the average error in the (new) image plane

[1]This data was kindly provided to us by Tim Child from TELEVIRTUAL.

was only 1.15 pixel. We attribute this to the fact that there is more variation in the side view comparing to the front view (Figure 3) and that the 3D hierarchical model has been trained on insufficient data, but this observation is worthy of further investigation.

When we used the second level of the hierarchy for fine-tuning we found that the results were not significantly different, and on occasion worse. We conjecture that results would improve were we to use HMMs at each node in the skeletal model.
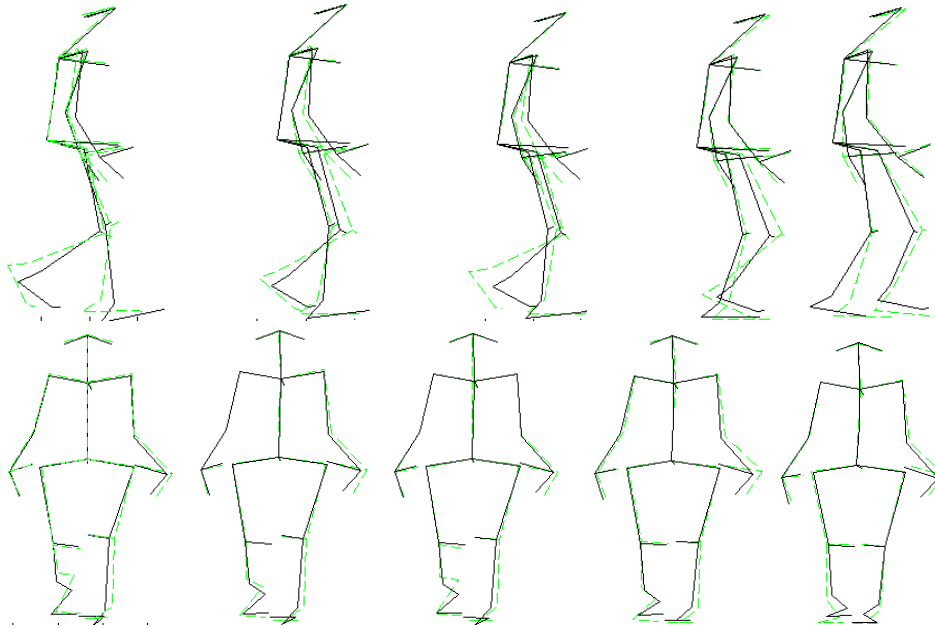

Figure 4: Tracking skeleton in 3D: front and recovered side views. The tracked figure is drawn with solid black lines and the ground truth figure is dashed grey

When tracking using the front view, the average error was about 5 and spread of 3 pixels, mainly due to a large degree of ambiguity of the front view. This ambiguity may be resolved in part by perspective projection, but also by appealing to additional information in the image such lighting information.

4.2 2D experiments

The training and testing data was obtained by hand-marking 24 video sequences of three people walking parallel to the camera field of view, each about 40 frames long, thus giving around 1000 frames altogether. The skeleton figure consists of 9 connected vertices representing the right side of the human body, right leg, right arm, right half of the torso and head positions (Figure 5).

The model was trained on 21 video sequences chosen from 24 that were available. It was tested on the remaining 3 videsequences of people it had been trained on, and also a video sequence of a person it had not been trained on.

The skeleton model consists of 2 levels, the first level being the for the right-hand side of the whole body and the second level consisting of three nodes, one for the right leg,

one for the right arm and one for the right part of the torso and head.

The average error for a person the model had been trained on is 2 pixels with a spread of about 0.5 pixel when using second level of the hierarchy and 3 pixels with a spread of about 0.5 pixel when using only the top level of the hierarchy, with the vertical size of the whole figure being about 400 pixels. This demonstrates an improvemnet. The average error for a person the model has not been trained on is 9 pixels, with a spread of 4 pixels when using the full hierarchy. For a whole body model alone the average error was about 15 pixels. Again, there are benefits to be had from the hierarchical model.
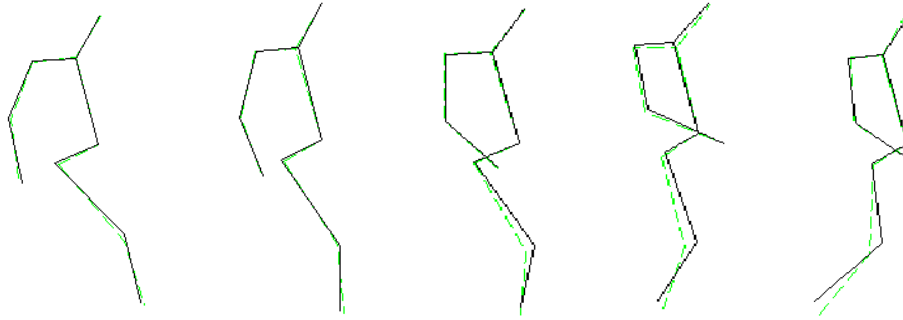

Figure 5: Tracking skeleton in 2D: The tracked figure is drawn with solid black lines and the ground truth figure is dashed grey

5 Conclusions and Future Research

We described a novel hierarchical model for view independent tracking of the human skeleton figure in monocular video sequences. The main contribution of our hierarchical model is the representation of minor variations of a 3D data set in a useful and compact manner, which allows greater specificity while tracking. We trained and tested the model on 3D data and showed that the system is capable of deriving 3D data from just one, not specified, 2D view.

We also trained the system on 2D data collected from the video sequences of 3 different people. The precision improved when we used the second level of hierarchy. The system was also able to track the 2D skeleton of a person it had not been trained on, thus showing that it is general enough to track different people, including previously not seen.

Our model is not homogeneous — HMM appears only at the root node. This may explain the deterioration of the performance in particular 3D situations when using the whole hierarchy. However, the 3D data was in insufficient quantity for us to be sure of our conclusions in this regard. Clearly, further work is needed. Nonetheless we were able to demonstrate view-independence using 3D data.

In 2D we tracked hand-segmented views, in future we will track automatically. Now that we have a model able to track people previously unseen, automatic tracking will be assisted.

In our future work we are also going to make our models extendable by building on our previous work in [7].

References

[1] C.Bregler and J.Malik. Tracking People with Twists and Exponential Maps. IEEE CVPR Proceedings, 1998. Also available at *http://www.cs.berkley/edu/~bregler/pubs.html*.

[2] R.Bowden, T.A.Mitchell, M.Sarhadi. Reconstructing 3D Pose and Motion from a Single Camera View. BMVC Proceedings, pp 904-913, 1998.

[3] J.Deller, J.Proakis and J.Hansen. Discrete-Time Processing of Speech Signals. Macmillan Publishing Company, 1993.

[4] L.Goncalves, E.Bernardo, E.Ursella and P.Perona. Monocular tracking of the human arm in 3D. ICCV Proceedings, pp 764-770, 1995.

[5] D.M.Gavrila and L.S.Davis. 3-D model-based tracking of humans in action: a multi-view approach. CVPR Proceedings, pp 73-79, 1996.

[6] D.Hogg. Model-based vision: a program to see a walking person. Image and Vision Computing, pp 5-20, February 1983.

[7] P.Hall, D.Marshall, R.Martin. Adding and Subtracting eigenspaces. BMVC Proceedings, pp 453-462, September1999.

[8] T.Heap and D.Hogg. Improving specificity in pdms using a hierarchical approach. BMVC Proceedings, pp 80-89, September 1997.

[9] M.Isard and A.Blake. Condensation-conditional density propagation for visual tracking. International J. Computer Vision, vol. 28, pp 5-28, 1998.

[10] P.H. Kelly, E.A.Hunter, K.Kreutz-Delgado and R.Jain. Lip Posture Estimation using Kinematically Constrained Mixture Models. BMVC Proceedings, pp 74-83, September 1998.

[11] I.A.Kakadiaris, D.Metaxas. Model-based Estimation of 3D Human Motion with Occlusion Based on Active multi-viewpoint selection. CVPR Proceedings, pp 81-87, June 1996.

[12] E.J.Ong and S.Gong. A dynamic Human Model using Hybrid 2D-3D Representations in Hierarchical PCA space. BMVC Proceedings, pp 33-42, September 1999.

[13] L.Rabiner and B.Juang. An Introduction to Hidden Markov Models. IEEE ASSP Magazine, pp 4-16, January 1986.

[14] T.Starner and A.Pentland. Real-Time American Sign Language Recognition from Video Using Hidden Markov Models. M.I.T. Media Laboratory Perceptual Computing Section Technical report No. 375, available at *http://vismod.www.media.mit.edu/cgi-bin/tr_pagemaker*.

[15] M.Walter, A.Psarrou and S.Gong. Learning Prior and Observation Augmented Density Models for Behaviour Recognition. BMVC Proceedings, pp 33-42, September 1999.

[16] A.Watt, 3D Computer Graphics. Addison-Wesley, third edition, pp 493-496, 2000.