Real-Time 3-D Human Body Tracking using Variable Length Markov Models

Fabrice Caillette Aphrodite Galata Toby Howard

Advanced Interfaces Group, School of Computer Science
University of Manchester, Manchester M13 9PL, UK
{ fabrice, a.galata, toby }@cs.man.ac.uk

Abstract

In this paper, we introduce a 3-D human-body tracker capable of handling fast and complex motions in real-time. The parameter space, augmented with first order derivatives, is automatically partitioned into Gaussian clusters each representing an elementary motion: hypothesis propagation inside each cluster is therefore accurate and efficient. The transitions between clusters use the predictions of a Variable Length Markov Model which can explain high-level behaviours over a long history. Using Monte-Carlo methods, evaluation of model candidates is critical for both speed and robustness. We present a new evaluation scheme based on volumetric reconstruction and blobs-fitting, where appearance models and image evidences are represented by Gaussian mixtures. We demonstrate the application of our tracker to long video sequences exhibiting rapid and diverse movements.

1 Introduction

Full human-body tracking has a wide and promising range of applications, from motion capture in the film industry to Human-Computer Interaction. Tracking people is difficult because of the high dimensionality of full body kinematics, the fast movements and frequent self-occlusions. Moreover, loose clothing, shadows or camera noise may further complicate the inference problem.

Tracking is a global optimisation process: because of kinematic constraints, even relatively independent limbs must compete to fit onto their own detected features (image evidence). Hierarchical methods [2, 16] fit the torso in a first stage and then optimise each limb independently. The parameter space is then partitioned, which drastically reduces the complexity of inference. However, problems occur when the torso cannot accurately be located on its own, which can be the case in human body tracking because of self-occlusions, or simply measurement noise.

One approach to tracking as a global optimisation problem is to start from image data, trying to detect features independently in each frame. The configuration of the model is then recovered from the "bottom-up" [17], using nonparametric belief propagation techniques. Since the feature detectors will inevitably return many false positives, the configuration of the model is globally optimised by iterating belief propagation in a graph with strong kinematic and temporal priors [20]. While these techniques are theoretically appealing, they rely on the detection of specific features, which is not always possible because of occlusions or loose clothing. Additionally, the computational complexity of the method is currently too high for real-time applications.

Alternatively, one can use the body configuration in the current frame and a dynamic model to predict the next configuration candidates (*motion prior*). These candidates are then tested against image data to find the most likely configuration. Tracking with particle filters works along those lines, approximating the *posterior* distribution by a set of representative elements, and updating these particles with Monte Carlo importance sampling rule [12]. However, in full body tracking problems, the dimensionality of the parameter space is far too high to represent accurately the true posterior distribution everywhere. Instead, particles tend to concentrate in only a few of the most significant modes, leading to possible failures when too few particles are propagated to represent a new peak. Annealing [6] is a coarse to fine approach that can help focus the particles on the global maxima of the posterior, at the price of multiple iterations per frame. Alternatively, sophisticated motion prior models have been proposed [19], trying to predict the subject's dynamics and propagating particles around the next expected peaks of the posterior.

Prediction is hard because human dynamics are complex and highly non-linear. Models of linear dynamics such as Kalman filters suffice to predict simple linear motions, but a better prediction model is required for faster and more complex movements. When the target motion is relatively short and structured, projecting the parameters onto a lower dimensionality manifold [13] encodes implicitly the correlations between parameters, and makes linear prediction methods efficient again. Such methods have shown to predict successfully walking cycles using Autoregressive Models [1]. Problems reappear with long sequences of complex motions, where the parameters are not sufficiently correlated to give good predictions under projection.

The main performance bottleneck when using Monte-Carlo methods is the evaluation of the likelihood function. For each particle, it usually involves generating a 3-D appearance model from the particle state, projecting this appearance model onto the available image planes, and finally comparing it with some extracted image features such as silhouettes or edges. Various simplifications or optimisations [4] have been attempted, but none of them were able to make full use of image information in real-time.

In this paper, we present novel prediction and evaluation schemes making robust tracking of challenging human motions possible in real-time. Prediction is based on behaviour models, capable of exploiting local dynamics as well as long history, whereas our new evaluation procedure, based on volumetric reconstruction and blobs-fitting, allows a large number of model candidates to be tested in a very efficient manner. The tracker is also able to recover from tracking failures by using the motion prototypes as new starting points.

In Section 2, we show how complex movements are decomposed into clusters of elementary motions, and how high-order behaviour is learnt over these clusters. The actual tracking is performed by a Sample Importance Resampling (SIR) particle filter [12], with a propagation of the particles following the dynamic model described in Section 4. A method for fast evaluation of the particles is then introduced 5. Section 6 and 7 respectively present some results and discussion.

2 Human body representation

In this section we describe the parametrisation we use for the human body as well as the features we use to learn the human behaviour model that will constrain the search within

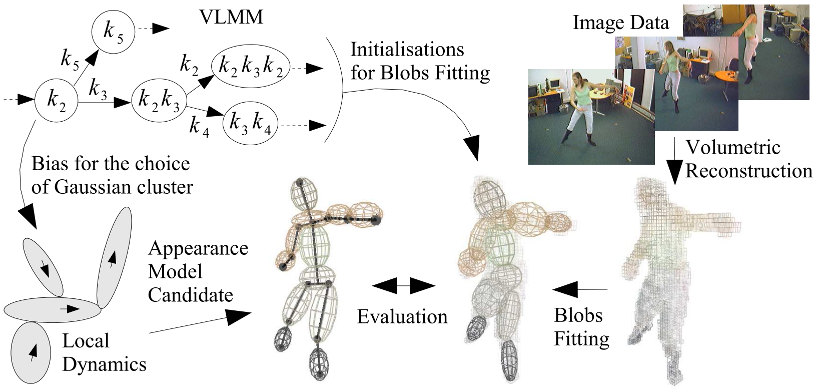
VLMM
k_5
k_5
k_2
k_3
k_2
$k_2 k_3 k_2$
$k_2 k_3$
k_4
$k_3 k_4$
Initialisations for Blobs Fitting
Image Data
Volumetric Reconstruction
Bias for the choice of Gaussian cluster
Appearance Model Candidate
Evaluation
Blobs Fitting
Local Dynamics

Figure 1: Overview of the system.

our proposed Bayesian tracking framework.

2.1 Kinematic Tree and Constraints

The model of the human body is based on a kinematic tree consisting of 14 segments, as seen in Figure 1. Each pose is represented by a 25-dimensional vector $\mathbf{C}_t$ which consists of the joint angles, and the position and orientation of the root of the kinematic tree.

Constraints are placed on joint rotations (expressed as Euler angles) in the form of bounding values. Redundant configurations and singularities are eliminated by limiting each joint to two degrees of freedom. The constraints restrict the number of impossible poses, but are insufficient to capture the complexity of human morphological constraints. More advanced constraints schemes have been proposed [15], but in our case, a high level behaviour model learnt from training sets of 3D human motions (e.g., joint angles over time) will implicitly play the same role.

2.2 Feature space representation

In order to learn a concise probabilistic model of 3D human motion, we need to choose an appropriate feature space. For each body pose, we define a corresponding feature vector $\mathbf{X}_t = (\mathbf{x}_t, \dot{\mathbf{x}}_t)$ consisting of the joint angles vector $\mathbf{x}_t$ and its first derivative $\dot{\mathbf{x}}_t$. Global position and orientation are omitted from the chosen feature representation as we do not wish the learnt behaviour model to be sensitive to them. The inclusion of derivatives helps resolve ambiguities in configuration space. Moreover, it facilitates the use of models in performing generative tasks using local dynamics (see Section 4.2).

Human body behaviour may be viewed as a smooth trajectory within the feature space that is sampled at frame rate, generating a sequence of feature vectors $\mathbf{X}_t$. Each sequence describes the temporal evolution of human body poses (augmented by the first derivatives of the joint angles): $\{\mathbf{X}_1,\ \mathbf{X}_2,\ \ldots,\ \mathbf{X}_m\}$.

3 Learning Dynamics

3.1 Clustering the Feature Space

Due to the complexity of human dynamics, we break down complex behaviours into elementary movements for which local dynamic models are easier to infer. The problem

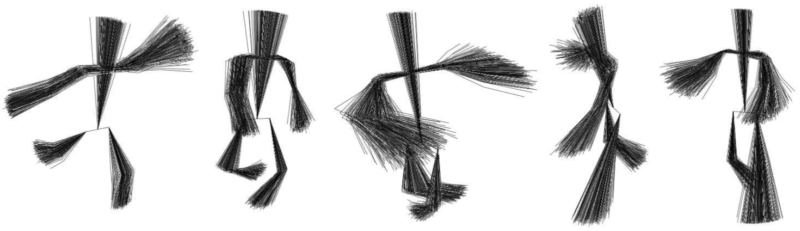

Figure 2: Model configurations sampled from various Gaussian clusters. Note that the derivatives are not shown, and the training data for head movements were not available.

is then to automatically find, isolate and model these elementary movements from the training data. We achieve this by clustering the feature space into Gaussian clusters using a variant of the EM algorithm proposed by Figueiredo and Jain [7]. Their proposed method automatically addresses the main pitfalls of traditional EM, that is, the delicate initialisation, the arbitrary choice of the number of components, and the possibility of singularities. Body configurations sampled from a few clusters on ballet-dancing data are shown in Figure 2.

3.2 Learning High-Level Behaviour with VLMMs

Complex human activities such as dancing (or even simpler ones such as walking), can be viewed as a sequence of primitive movements with a high level structure controlling the temporal ordering.

By incorporating probabilistic knowledge of the underlying behavioural structure in the way we sample our particles (in a Bayesian tracking framework using Monte Carlo simulation), we can propagate particles only in plausible directions, and also provide automatic transitions between the different model configurations. A suitable way to obtain such knowledge is variable-length Markov models (VLMMs) [18].

Variable length Markov models deal with a class of random processes in which the memory length varies, in contrast to an n-th order Markov models. They have been previously used in the data compression [5] and language modelling domains [18, 14]. More recently, they have been successfully introduced in the computer vision domain for learning stochastic models of human activities with applications to behaviour recognition and behaviour synthesis [9, 10, 8]. Their advantage over a fixed memory Markov model is their ability to locally optimise the length of memory required for prediction. This results in a more flexible and efficient representation which is particularly attractive in cases where we need to capture higher-order temporal dependencies in some parts of the behaviour and lower-order dependencies elsewhere. A detailed description on building and training variable-length Markov models is given by Ron *et al.* [18].

A VLMM can be thought of as a probabilistic finite state automaton (PFSA) $\mathscr{M} = (Q, \Sigma, \tau, \gamma, s)$, where Σ is a set of tokens that represent the finite alphabet of the VLMM, and Q is a finite set of model states. Each state corresponds to a string in Σ of length at most $N_{\mathscr{M}}$ ($N_{\mathscr{M}} \geq 0$), representing the memory for a conditional transition of the VLMM. The transition function τ, the output probability function γ for a particular state, and the probability distribution s over the start states are defined as:

$$\tau : Q \times \Sigma \rightarrow Q \qquad \gamma : Q \times \Sigma \rightarrow [0,1] \qquad s : Q \rightarrow [0,1]$$

The VLMM is a generative probabilistic model: by traversing the model's automaton

$\mathscr{M}$ we can generate sequences of the tokens in Σ. By using the set of Gaussian clusters as the alphabet, we can capture the temporal ordering and space constraints associated with the primitive movements. Consequently, traversing $\mathscr{M}$ will generate statistically plausible examples of the behaviour.

4 Predictions using the Dynamic Model

Using Bayes' rule, the probability of a model configuration $\mathbf{x}_t$ given a measurement $\mathbf{z}_t$ is:

$$\underbrace{P(\mathbf{x}_t|\mathbf{Z}_t)}_{Posterior} = \kappa.\underbrace{P(\mathbf{z}_t|\mathbf{x}_t)}_{Likelihood}.\int \underbrace{P(\mathbf{x}_t|\mathbf{x}_{t-1})}_{Motion\,Prior}.\underbrace{P(\mathbf{x}_{t-1}|\mathbf{Z}_{t-1})}_{Previous\,posterior}\,d\mathbf{x}_{t-1} \tag{1}$$

where κ is a normalising constant, and $\mathbf{Z}_t = \{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_t\}$. The *posterior* distribution is approximated by a set of discrete particles, each representing a body configuration. In this section, we shall describe a behaviour-based *motion prior* using the VLMM for prediction. A fast way of evaluating the *likelihood* using volumetric reconstruction and blobs fitting will then be presented in Section 5.

4.1 Transitions Between Clusters with the VLMM

The particles are augmented with their current VLMM state q_t, from which the cluster k_t they belong to is easily deduced. Transitions (or jumps) between clusters are conditional on the particle's feature vector $\mathbf{X}_t$ as well as the transition probabilities γ in the VLMM. The probability of transition towards a new Gaussian cluster k_{t+1} of mean $\mu_{k_{t+1}}$ and covariance $\Sigma_{k_{t+1}}$ is:

$$\begin{aligned} P(k_{t+1} \mid \mathbf{X}_t, q_t) &\propto P(\mathbf{X}_t \mid k_{t+1}).P(k_{t+1} \mid q_t) \\ &= \frac{1}{\sqrt{(2\pi)^d \left|\Sigma_{k_{t+1}}\right|}}.e^{-\frac{1}{2}.(\mathbf{X}_t - \mu_{k_{t+1}})^T \cdot \Sigma_{k_{t+1}}^{-1} \cdot (\mathbf{X}_t - \mu_{k_{t+1}})}.\gamma(q_t, k_{t+1}) \end{aligned} \tag{2}$$

At each frame, the state transition is chosen according to the above probabilities for each neighbouring cluster. In practice, only a few transitions are encoded in the VLMM, making the evaluation efficient. If the same cluster is chosen ($k_{t+1} = k_t$), the particle is propagated using local dynamics, as formulated in the next section. If a new cluster is selected, the particle's parameters are re-sampled from the new Gaussian cluster.

4.2 Local Dynamics

Inside each Gaussian cluster, a new model configuration can be stochastically predicted from the previous feature vector $\mathbf{X}_t$. Since the Gaussian clusters include derivatives, the prediction effectively behaves like a second-order model. Let us consider a Gaussian cluster of mean $\mu = \left(\begin{smallmatrix} \mu_X \\ \mu_{\dot{X}} \end{smallmatrix}\right)$ and covariance matrix $\Sigma = \left(\begin{smallmatrix} \Sigma_{XX} & \Sigma_{X\dot{X}} \\ \Sigma_{X\dot{X}}^T & \Sigma_{\dot{X}\dot{X}} \end{smallmatrix}\right)$. The noise vector is directly sampled from the cluster's covariance matrix with an attenuation coefficient λ, leading to the formulation:

$$\begin{array}{rcl} \dot{\mathbf{x}}_t &=& \dot{\mathbf{x}}_{t-1} + \lambda.d\dot{\mathbf{x}}_t \\ \mathbf{x}_t &=& \mathbf{x}_{t-1} + \dot{\mathbf{x}}_t + \lambda.d\mathbf{x}_t \end{array} \qquad \text{with} \qquad \begin{pmatrix} d\mathbf{x}_t \\ d\dot{\mathbf{x}}_t \end{pmatrix} \sim \mathscr{N}(0, \Sigma) \tag{3}$$

The random noise vector is drawn as $\left(\begin{smallmatrix} d\mathbf{x}_t & d\dot{\mathbf{x}}_t \end{smallmatrix}\right)^T = \sqrt{\Sigma} \cdot X$ with $X \sim \mathscr{N}(0, I)$. The square-root of the covariance matrix is computed by performing the eigenvalue decomposition,

$\Sigma = V \cdot D \cdot V^T$, and taking the square root of the eigenvalues on the diagonal of D, so that $\sqrt{\Sigma} = V \cdot \sqrt{D} \cdot V^T$.

This predictive model has to be understood in the context of Monte-Carlo sampling, where noise is introduced to model uncertainty in the prediction: the properties of the noise vector are therefore almost as important as the dynamics themselves. The covariance matrix of the current cluster provides a good approximation of this uncertainty, and sampling the noise vector from the cluster itself makes propagation of uncertainty much closer to the training data than uniform Gaussian noise.

To keep the behaviour model independent of the global position and orientation of the subject, the six global parameters are not modelled by the Gaussian clusters, and are therefore propagated with a uniform noise.

5 Fast Evaluation of the Likelihood

5.1 Appearance Model

Appearance is modelled by 3-D blobs attached along the bones of the kinematic model. The shape of a blob is described by a Gaussian distribution of mean μ_X and covariance matrix Σ_X. Since the blobs are generated in the local coordinate system of each body part, we retain only four free parameters: a single offset value which summarises the mean μ_X along the first axis of the bone on which the blob is attached, and the three eigenvalues which fully describe the covariance matrix Σ_X. The transformation needed to convert blobs from local to global coordinates is obtained using forward kinematics.

Blobs also incorporate colour information which, similarly to shape, is represented by a Gaussian distribution of mean μ_C and covariance matrix Σ_C. The full blob parameters are learnt automatically during the first seconds of the tracking using Expectation-Maximisation on the voxel data (see Section 5.3).

Since the colour of each blob is unimodal, clothing with multiple colours must be handled by a mixture of blobs. Starting with a single blob for each body-part, a “split and merge” process ensures an optimal description of the data. The criterion used to decide whether a blob should be split is the colour variance along the main spatial axis of the blob. This measurement is obtained by projecting the mixed covariance matrix between spatial and colour information Σ_{XC} (computed from the data with EM) onto the direction of the current bone in the kinematic model.

5.2 Volumetric Reconstruction

Volumetric reconstruction has the advantage of combining relevant information for tracking (shape and colour) into a single coherent structure. Although other features like edges or texture can provide valuable information, the unavoidable motion blur hinders their robustness when dealing with fast motions. We argue that shape-from-silhouette algorithms, by exploiting correspondences between camera views, can yield more robustness and performance than individual image-based feature extraction. A real-time hierarchical method for voxel-based volumetric reconstruction has been the subject of our previous work [2]. In this work, using calibrated cameras, the visual-hull algorithm projects 3-D voxels onto available image planes and keeps those which lie inside all the silhouettes of the object of interest. Our contribution consists in merging silhouette extraction and volumetric reconstruction into a hierarchical scheme, which has the double advantage of robust pixel statistics and improved performance. Colour information is also recovered, making the reconstructed volume a valuable basis for tracking.

5.3 Data Density as a Mixture of Gaussians

The volumetric reconstruction summarises the data by keeping only relevant information (shape and colour). Unfortunately, the amount of data is still too large for real-time evaluation of candidate configurations, therefore a more compact representation of the data is needed. In [2] and [3], we also showed how to fit a mixture of Gaussian blobs onto the 3-D voxels in real-time, using an EM-like procedure. Provided that this blob-fitting procedure is reliable enough, it constitutes an ideal basis for efficient evaluation of particles.

Just like for every EM-based algorithm, the reliability of blob-fitting strongly depends on initialisation. The number of blobs and their attributes are known from the appearance model, but their actual positions depend on the pose of the underlying kinematic model. Initialising EM from the tracked position in the last frame can prove insufficient for fast movements. Fortunately, the VLMM can predict the next possible clusters by traversing the automaton from the last tracked position. EM is then performed from the centres of these clusters, and the maximum-likelihood result is retained.

This blobs-fitting procedure has the important advantage of detecting tracking failures: if the best mixture has a low likelihood, the tracker is lost and needs re-initialisation. Unlike most other trackers, automatic recovery from failures is then possible because the parameter space is clustered in motion prototypes. Performing EM from all clusters might provoke a noticeable lag, depending on the total number of prototypes, but is bound to return a good result. The VLMM state of all particles is then reset, which has the effect of spreading them across the clusters. To ensure a quick recovery, a bias towards the clusters that returned the best mixtures is introduced for the first state transition (Section 4.1).

5.4 Particle Evaluation

A model configuration (particle) is evaluated by first generating an appearance model from the particle state, and then comparing the produced blobs with those obtained from the image evidence. Let us note $F = \sum_i \alpha_i f_i$ the mixture generated from the model and $G = \sum_i \beta_i g_i$ the one corresponding to image evidences. The Kullback-Leibler (KL) divergence can be used to measure the cross-entropy between the two mixtures:

$$D_{KL}(F\|G) = \int F \ln \frac{F}{G} = \sum_i \alpha_i \int f_i \ln F - \sum_i \alpha_i \int f_i \ln G \tag{4}$$

Using the approximation proposed by [11] for non-overlapping clusters:

$$\begin{aligned} D_{KL}(F\|G) &\simeq \sum_i \alpha_i \int f_i \ln \alpha_i f_i - \sum_i \alpha_i \max_j \int f_i \ln \beta_j g_j \\ &= \sum_i \alpha_i \min_j (D_{KL}(f_i\|g_j) + \ln \frac{\alpha_i}{\beta_j}) \end{aligned} \tag{5}$$

Correspondence between blobs is maintained under the form $f_i \leftrightarrow g_{\pi(i)}$, so that the complexity of the run-time evaluation function is linear with respect to the number of blobs:

$$D_{KL}(F\|G) \simeq \sum_{i=1}^{n} \alpha_i \left(D_{KL}(f_i\|g_{\pi(i)}) + \ln \frac{\alpha_i}{\beta_{\pi(i)}} \right) \tag{6}$$

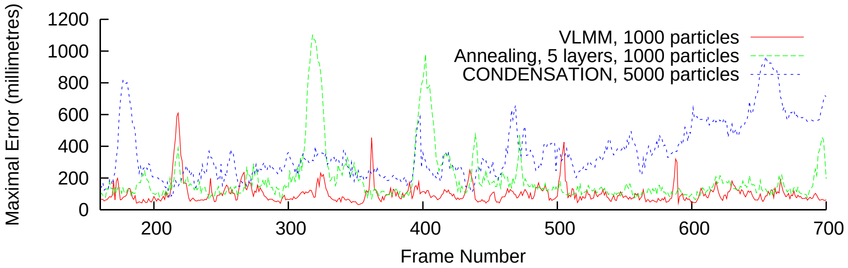

Figure 3: Maximum distance between the tracked joint locations and the ground-truth data on a ballet dancing test sequence (not included in the training data). The average error over the sequence is 36mm for CONDENSATION, 16mm for the Annealed particle filter, and 13mm for our method.

This last formulation can be efficiently computed using the closed form solution of the KL divergence between two Gaussian blobs $f \sim \mathcal{N}(\mu_f, \Sigma_f)$ and $g \sim \mathcal{N}(\mu_g, \Sigma_g)$:

$$D_{KL}(f\|g) = \frac{1}{2}\left(\ln\frac{|\Sigma_f|}{|\Sigma_g|} - d + tr(\Sigma_f^{-1}\Sigma_g) + (\mu_g - \mu_f)^T\Sigma_f^{-1}(\mu_g - \mu_f)\right) \tag{7}$$

where d is the dimensionality of the Gaussian blobs f and g.

6 Results

Our novel prediction and evaluation methods were tested on long video sequences exhibiting fast and diverse movements. Ballet dancing is an interesting application because movements are so fast that, at normal framerate, tracking without an adequate dynamic model is very challenging. The volumetric reconstruction is based on 4 cameras, capturing images at 30fps in a resolution of 320×240.

Our training data consisted of 8 sequences of ballet-dancing motion capture, approximately 2000 frames each. When partitioning the parameter-space, the optimal number of clusters was automatically found to be 256, which can seem quite high but actually reflects the underlying complexity of the motions. As a comparison, the same clustering on a simpler “arms pointing” sequence returned only 5 clusters. We then learnt a VLMM over the Gaussian clusters using various memory lengths. Note that a memory length of 1 makes the VLMM behave like a first order Markov model. Using a memory length of 5, the VLMM learnt 734 distinct states. This number of states rose to 1722 with a memory length of 10. To avoid overfitting which leads to poor performance when encountering unseen events, a maximum memory length of 5 was chosen.

A comparison of accuracy between our algorithm and other standard particle filter methods can be found in Figure 3. The CONDENSATION algorithm propagates particles with a Gaussian noise, while Annealing [6] iterates a propagation-evaluation loop over multiple layers, in a “coarse to fine” manner. Even using 5000 particles, CONDENSATION was unable to explore the parameter-space in all appropriate directions, resulting in a rapid failure of tracking. The Annealed particle filter uses only 1000 particles, but because of the 5 layers of annealing, the computational cost remains equivalent to CONDENSATION. Annealing produces accurate results in most of the test sequence, although some tracking failures still occur because of the relatively low number of particles. Despite having 5 times less particle-evaluations than the two other methods, our

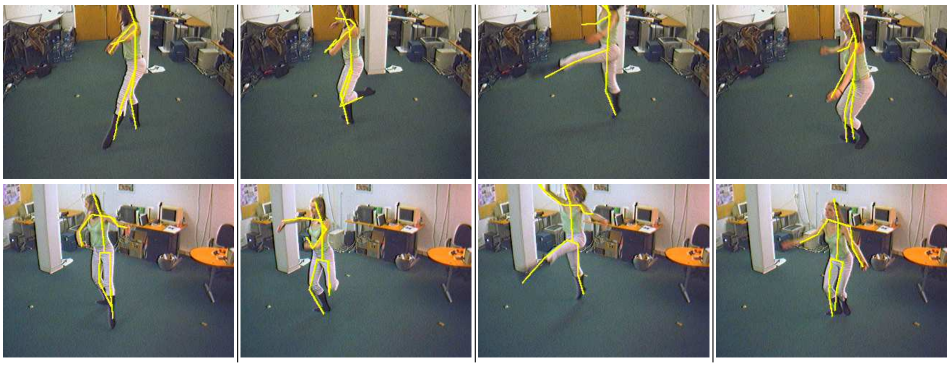

Figure 4: Tracking ballet dancing movements (two camera views shown).

propagation scheme maintains accuracy and robustness. Occasional tracking failures, due to movements unseen in the training set, are detected and quickly recovered from.

Visual tracking results are presented in Figure 4. Motion blur and the cluttered background make the reconstruction challenging, but the motion model copes with incomplete data. The full system, which comprises image acquisition, volumetric reconstruction and the Bayesian tracking framework runs at 10fps with a pool of 1000 particles on a single 2GHz computer.

7 Discussion

The main challenge in human-body tracking is the high dimensionality of the parameter space, making the search for the correct pose a hard problem. Using Monte-Carlo methods, the number of required particles tends to become very large, and even if methods such as Annealing improve convergence, the computational cost remains too high for real-time applications.

In this paper, we have demonstrated an algorithm using high-level behaviours to track challenging movements in real-time. Novel contributions reside in the prediction scheme which uses VLMMs and in a fast evaluation method based on volumetric reconstruction and blobs fitting. By focusing the propagation of particles towards predicted directions, the number of particles required for robust tracking is kept low, and in conjunction with a fast evaluation scheme, real-time performance is then achieved on commodity hardware.

As future research directions, we intend to investigate and evaluate various dimensionality reduction methods, in an effort to make the learning of clusters more efficient. Online learning, where unseen sequences are incrementally integrated into the behaviour model, would also represent a worthy contribution.

Acknowledgements

We would like to thank Dr Kia Ng and the Interdisciplinary Centre for Scientific Research in Music at the University of Leeds for allowing the use of their motion capture facilities. We would also like to thank Mr Royce Neagle for the choreography and performance of the training ballet sequences and for his technical help in capturing the motion data. Finally, we would like to thank Kate Simmons Dance Ltd.

References

[1] A. Agarwal and B. Triggs. Tracking articulated motion using a mixture of autoregressive models. In *Proc. ECCV*, volume 3023, pages 54–65, 2004.

[2] F. Caillette and T. Howard. Real-Time Markerless Human Body Tracking with Multi-View 3-D Voxel Reconstruction. In *Proc. BMVC.* vol. 2, pp. 597–606, 2004.

[3] F. Caillette and T. Howard. Real-Time Markerless Human Body Tracking Using Colored Voxels and 3-D Blobs. In *Proc. ISMAR*, pages 266–267, Nov. 2004.

[4] J. Carranza, C. Theobalt, M. Magnor, and H. Seidel. Free-viewpoint video of human actors. In *ACM Trans. Graph. (Proc. SIGGRAPH)*, pages 569–577, 2003.

[5] G. Cormack and R. Horspool. Data Compression using Dynamic Markov Modelling. *Computer Journal*, 30(6):541–550, 1987.

[6] J. Deutscher, A. Blake, and I. D. Reid. Articulated body motion capture by annealed particle filtering. In *Proc. CVPR*, volume 2, pages 126–133, 2000.

[7] M. A. T. Figueiredo and A. K. Jain. Unsupervised learning of finite mixture models. *IEEE Trans. on PAMI*, 24(3):381–396, 2002.

[8] A. Galata, A. G. Cohn, D. Magee, and D. Hogg. Modeling interaction using learnt qualitative spatio-temporal relations and variable length markov models. In *Proc. European Conference on Artificial Intelligence (ECAI'02)*, pages 741–745, 2002.

[9] A. Galata, N. Johnson, and D. Hogg. Learning Behaviour Models of Human Activities. In *Proc. BMVC*, pages 12–22, 1999.

[10] A. Galata, N. Johnson, and D. Hogg. Learning Variable Length Markov Models of Behaviour. *Computer Vision and Image Understanding*, 81(3):398–413, 2001.

[11] J. Goldberger, S. Gordon, and H. Greenspan. An efficient image similarity measure based on approximations of KL-divergence between two gaussian mixtures. In *Proc. ICCV*, pages 487–493, 2003.

[12] N. Gordon, J. Salmond, and A. Smith. Novel approach to non-linear/non-gaussian bayesian state estimation. In *Radar and Signal Processing*, pages 107–113, 1994.

[13] K. Grochow, S. L. Martin, A. Hertzmann, and Z. Popovic. Style-based inverse kinematics. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 23(3):522–531, 2004.

[14] I. Guyon and F. Pereira. Design of a Linguistic Postprocessor using Variable Memory Length Markov Models. In *ICDAR*, pages 454–457, 1995.

[15] L. Herda, R. Urtasun, and P. Fua. Hierarchical implicit surface joint limits to constrain video-based motion capture. In *Proc. ECCV*, volume 2, pages 405–418, 2004.

[16] I. Mikic, M. Trivedi, E. Hunter, and P. Cosman. Human body model acquisition and tracking using voxel data. *IJCV*, 53(3):199–223, 2003.

[17] D. Ramanan and D. A. Forsyth. Finding and tracking people from the bottom up. In *Proc. CVPR*, volume 2, pages 467–475, 2003.

[18] D. Ron, Y. Singer, and N. Tishby. The power of amnesia: Learning probabilistic automata with variable memory length. *Machine Learning*, 25(2–3):117–149, 1996.

[19] H. Sidenbladh, M. J. Black, and L. Sigal. Implicit probabilistic models of human motion for synthesis and tracking. In *ECCV*, volume 1, pages 784–800, 2002.

[20] L. Sigal, S. Bhatia, S. Roth, M. J. Black, and M. Isard. Tracking loose-limbed people. In *Proc. CVPR*, volume 1, pages 421–428, 2004.