Tracking-as-Recognition for Articulated Full-Body Human Motion Analysis

Patrick Peursum Svetha Venkatesh Geoff West
Dept of Computing, Curtin University of Technology GPO Box U1987, Perth, Western Australia
{P.Peursum, S.Venkatesh, G.West}@curtin.edu.au

Abstract

This paper addresses the problem of markerless tracking of a human in full 3D with a high-dimensional (29D) body model. Most work in this area has been focused on achieving accurate tracking in order to replace marker-based motion capture, but do so at the cost of relying on relatively clean observing conditions. This paper takes a different perspective, proposing a body-tracking model that is explicitly designed to handle real-world conditions such as occlusions by scene objects, failure recovery, long-term tracking, auto-initialisation, generalisation to different people and integration with action recognition. To achieve these goals, an action's motions are modelled with a variant of the hierarchical hidden Markov model. The model is quantitatively evaluated with several tests, including comparison to the annealed particle filter, tracking different people and tracking with a reduced resolution and frame rate.

1. Introduction

A variety of approaches to the problem of markerless 3D full-body human motion capture have been proposed in the literature. Lee *et al*. [7] and Mikić *et al*. [8] both constrain the possible posture configurations by analytically finding the hands, face and/or torso. Lee then transitions a particle filter under these constraints while Mikić 'grows' the body-part tree to best fit the voxel-based visual hull observation. These both require reliable analytical detection that is difficult to guarantee. Deutscher *et al*. [5] propose the annealed particle filter (APF) that uses simulated annealing to gradually focus the search effort on promising areas. The algorithm is effective but tends to converge on only one mode, discarding the rest of the posture distribution. Sminchisescu and Jepson [13] explicitly maintain multi-modality by using a combination of kinematic jumps, sampling and variational methods to track and smooth multiple plausible posture trajectories. Their system is able to recover an accurate 3D posture sequence with only a monocular view, (albeit producing many other posture trajectories at the same time) at the expense of a complex, multi-layered algorithm structure and an implicit reliance on a close-fitting body model. Recent work by Caillette *et al*. [4] learns Gaussian clusters of sub-motions and trains a variable-length Markov model (VLMM) based on these clusters to direct the local posture search towards better areas of the distribution. They achieve near-real-time (10fps) performance with a visual hull on long video sequences of a ballet dancer who stays in a relatively fixed location. Their algorithm auto-initialises and can recover from errors but it is tightly integrated with the visual hull, which requires many views and can be sensitive to segmentation errors.

All of these algorithms aim to be replacements for marker-based motion capture. However, for applications that wish to build on top of 3D full-body tracking in real-world situations there are certain complicating factors that are not present in – nor even a concern of – pure motion capture. These include (a) significant observation errors; (b) reliable long-term tracking; (c) automatic detection and initialisation; (d) tracking different people without fine-tuning a close-fitting body model for each; and (e) facilitating higher-level tasks such as action recognition.

Foremost are (a) significant observation errors caused by cluttered environments, occlusions and low-level algorithm failures. These are inevitable when working with real-world scenes, but are generally disregarded by 3D body trackers in order to make high-quality tracking attainable. Unfortunately, when observation errors occur the correct posture becomes *less* likely than other seemingly plausible postures and tracking failures become almost certain. This leads to issues with tracking over longer periods (b) — a failure in body tracking is often considered terminal because it skews the local search for the next posture into poor areas of the posture space. This brings into doubt the feasibility of long-duration tracking, especially given that most tracking results have been based on extremely short sequences (five seconds or less [5, 13]). In addition, body trackers typically begin with a perfect (manually-set) initialisation that ensures an ideal start to tracking. Automatic initialisation (c), mandatory for realistic deployment, cannot provide such an ideal start. Of note then is work by Caillette *et al*. [4], who learn a motion model that supports auto-initialisation and long-term tracking by re-initialising after every failure. However, they still rely on clean observations due to their use of a visual hull. Related to initialisation is the implicit reliance of most body trackers on a close-fitting body model (d). This is impractical when seeking to track different people since prior knowledge of each person's physique is not usually available in realistic applications. Finally, the issues

(a–d) inherent in real-world scenes can easily cause the output of full-body trackers to be inconsistent between videos of the same motion, making higher-level tasks like action recognition (e) much more difficult.

In essence, existing full-body trackers are less than ideal for use in real-world tracking and action recognition simply because they do not aim to fulfill these roles. Hence this paper proposes a 3D full-body tracker that is explicitly designed to both handle observation errors and ease the task of action recognition. From the perspective of action recognition, motion is considered the execution of a particular action. The proposed model takes advantage of this to use the action as a context that guides motion tracking, hence reducing the reliance on the error-prone observations and thereby improving tracking reliability. Tracking and recognition thus occur simultaneously – for a test sequence, the most likely model is the action label and this model's state sequence is the motion tracking. Due to the high-dimensional state space, each action is broken down into a two-level hierarchy of phases (sub-actions) and motion within each phase. The hierarchy is tractably modelled with a hierarchical hidden Markov model (HHMM) [3] by factoring the states of the lower level (which model the actual pose). Each action is then modelled by a different instance of this factored-state HHMM (FS-HHMM), and the most likely model for a given sequence provides the action label and posture sequence.

The basic approach of the FS-HHMM to tracking is similar to Zhao and Nevatia's 'tracking-as-recognition' concept [14]. However, they combine tracking and recognition by matching optic flow against labelled motion templates and filter (track) with an HMM to produce a maximum-likelihood sequence of motion. Also, template matching is best suited to their far-field low-detail views, where tracking individual features is infeasible. In contrast, the inverse is true with this paper's more detailed views – many useable features can be detected and tracked whereas reliably performing full-body posture recognition from a single frame is difficult. The FS-HHMM also has some similarities to the model of [4], where a VLMM of motion is used to perform auto-initialisation and tracking. However, their focus is on accurate motion capture and the consequent reliance on clean observations means that they do not consider handling messy scenes or recognition tasks.

This paper demonstrates that modelling motion in the context of action recognition provides many benefits to tracking in realistic conditions. Significant benefits of the approach include the ability to:

(1) auto-initialise without any user input;
(2) track through partial occlusions of a mobile person;
(3) auto-recover from failures;
(4) track *different* people without adjustments/retraining;
(5) robust to low resolutions and reduced frame rates;
(6) facilitate action recognition; and
(7) track based on very little training data.

The approach is quantitatively evaluated against four actions (walking, sitting down, standing up and opening a bar-fridge) and comparisons are made with the APF [5]. Particle filtered inference is used to tractably explore the posture distribution, and the FS-HHMM is shown to require only 1,000 particles for successful tracking.

Note that in this paper the experimental focus is on the tracking results. Some action recognition results are provided to present the ability of the FS-HHMM in this area, but a rigorous and comparative evaluation of action recognition with the FS-HHMM will be the topic of future work.

2. The FS-HHMM Model

2.1. Body Model and Observation Function

Body Model This paper employs a simple 29-dimensional model of the human body (Figure 1a), parameterised by 24 joint rotations and five global variables (x,y,z, orientation, scale). Section 2.3 describes how these parameters are auto-initialised. Scale applies to the entire body model since the relative length of each limb is fixed. Each body part is modelled with a cylinder whose sides are projected onto the 2D image and then joined with lines to produce the cardboard look for efficient projection [12]. The model is fairly loose-fitting so that any tracker based on it should generalise well to different people.

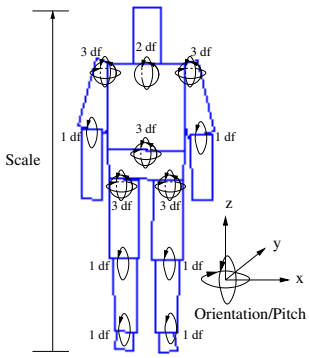


Figure 1. 29D 'cardboard' body model.

Observation Likelihood Function Given an observation y_t and a posture represented by the joint angles x_t, most 3D body trackers evaluate a function $f(y_t|x_t)$ to find the optimal posture x_t. Evaluating $f(y_t|x_t)$ is typically the most costly part of full-body tracking, and y_t is usually either a visual hull [4, 8] or foreground and edge images [5, 13], and $f(y_t|x_t)$ is a heuristic distance measure between y_t and x_t. Visual hull methods tend to be faster but inherit the visual hull's sensitivity to segmentation errors. In contrast, foreground/edge images require projecting x_t and expensive pixel-level evaluations for each view, but is more robust to segmentation errors in any one view. Since this paper aims to handle errors, it employs the latter approach:

$$f(y_t|x_t) = \frac{1}{\lambda}\exp\{\lambda \cdot Dist(y_t, Proj(x_t))\} \qquad (1)$$

where $Dist(\cdot)$ is a modified version of Deutscher's [5] cost function (as described in [10]) for the distance between y_t and x_t's projection and λ controls how sharply the distribution drops off with distance. For the FS-HHMM λ is fixed with λ=8 (chosen empirically). For the APF λ is varied dynamically as part of its annealing procedure.

2.2. Graphical Structure and Parameterisation

Figure 2 shows the FS-HHMM. Its parameters are:

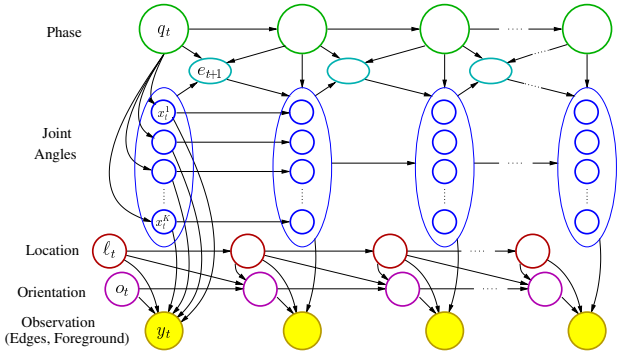


Figure 2. DBN of the factored-state HHMM

$$C_{mn} \triangleq P(q_t{=}n|q_{t-1}{=}m) \tag{2a}$$
$$A^k_{nij} \triangleq P(x^k_t{=}j|x^k_{t-1}{=}i, q_t{=}n, e^k_t{=}0) \tag{2b}$$
$$\Lambda^k_{nj} \triangleq P(x^k_t{=}j|q_t{=}n, e^k_t{=}1) \tag{2c}$$
$$\phi_m \triangleq P(q_1{=}m) \tag{2d}$$
$$\pi^k_{mi} \triangleq P(x^k_1{=}i|q_1{=}m) \tag{2e}$$
$$\Psi \triangleq P(\ell_t|\ell_{t-1}) \tag{2f}$$
$$\Upsilon \triangleq \omega^{o|o}_t P(o_t|o_{t-1}) + \omega^{o|\ell}_t P(o_t|\ell_{t-1}, \ell_t) \tag{2g}$$

where q_t is the phase, x_t is a 24D state modelling the body's joint rotations and is fully factored into 24 independently-transitioning sub-nodes x^k_t, $\{\ell_t, o_t\}$ is the location and orientation of the person in the scene and e_t controls phase transitions. The observation probability is modelled with a heuristic function $f(y_t|x_t, \ell_t, o_t)$. The FS-HHMM follows fairly intuitive mechanics and a simplified form can better illustrate the thinking behind it. Consider Figure 3, which for clarity drops e_t and merges related nodes together. This highlights the two-level state structure where each phase (cohesive sub-motion) is broken down into a set of joint configurations (postures) x_t within that phase. The body posture is then positioned in the scene by $\{\ell_t, o_t\}$.

In order to facilitate tractable inference, x_t is fully-factored. Strictly speaking, a 3df joint's three rotations should not be factored, but the assumption reduces the sparsity of the transition matrix and increases the possible set of postures (due to joint combinatorics) when training with a small data set. Each x^k_t discretises the 360° rotation space into 120 intervals of 3°. Discrete states are used for the sake of simplicity — a multinomial distribution can model non-linear transitions without the need for a non-linear mapping of the motion onto a lower-dimensional manifold. The $\{q_t, x_t\}$ hierarchy has some parallels with [4], which models a continuous posture space with a discrete set of Gaussian clusters. Their clusters are roughly equivalent to q_t, and samples within a cluster correspond to x_t. However, [4] does not discuss how internal transitions within a cluster (*i.e.* x_t transitions) are carried out.

Equations (2f) and (2g) model the movement of the overall body in the scene. Location Ψ (2f) is modelled as a linear-dynamics system. In contrast, orientation Υ (2g) is modelled by a two-component mixture – a linear-dynamics system $P(o_t|o_{t-1})$ from the previous orientation and a discrete distribution $P(o_t|\ell_{t-1}, \ell_t)$ learned from the training data and based on the person's direction of motion (change in location $\Delta\ell$). Note that this $P(o_t|\Delta\ell)$ is a discrete density to allow for uniform or multi-modal distributions. The rationale behind this mixture is that $P(o_t|o_{t-1})$ can become stuck in a poor orientation if (say) a walking person turns sharply. Since $\Delta\ell$ often has a relationship with o_t (*e.g.* a person walks in the direction they are oriented) samples from $P(o_t|\Delta\ell)$ can provide better orientations – essentially playing a self-correcting role in the tracking model. Note that $\omega^{o|o}_t$ and $\omega^{o|\ell}_t$ cannot be learned via EM because o_t is fully observed during training (see Section 2.3). These values are thus defined according to the following line of reasoning. Some actions (*e.g.* sitting) have no motion and so $\Delta\ell$ has no meaning for o_t. In such a case, $P(o_t|\Delta\ell)$ will be uniform and should not contribute to the mixture Υ. For actions with a *non*-uniform response to $\Delta\ell$, if the expectations $\mathbb{E}\big\langle P(o_t|o_{t-1})\big\rangle$ and $\mathbb{E}\big\langle P(o_t|\Delta\ell)\big\rangle$ closely agree, there is no reason to sample from the correcting distribution $P(o_t|\Delta\ell)$ since it is far noisier than $P(o_t|o_{t-1})$. On the other hand, if the two distributions are diametrically opposed (*e.g.* differ by 180°) then it is probable (but not certain) that a correction is needed. Thus $\omega^{o|o}_t$ and $\omega^{o|\ell}_t$ are calculated as:

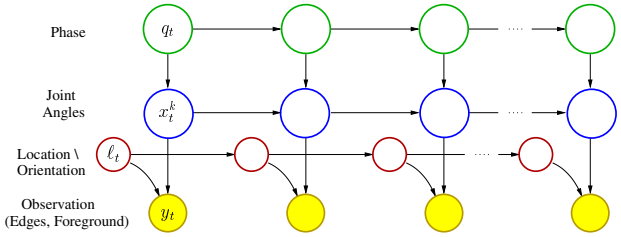


Figure 3. Simplified form of the FS-HHMM

$$u^{o|\ell}_t = \frac{1}{N\sum_n P(o_t{=}n|\ell_{t-1}, \ell_t)^2}, \quad u^{o|o}_t = \frac{1}{N\sum_n P(o_t{=}n|o_{t-1})^2} \tag{3a}$$
$$\widetilde{\omega}^{o|\ell}_t = \frac{1 - u^{o|\ell}_t}{\sum_z 1 - u^z_t} \tag{3b}$$
$$\omega^{o|\ell}_t = \widetilde{\omega}^{o|\ell}_t \cdot \left| \frac{\mathbb{E}\big\langle P(o_t|o_{t-1})\big\rangle - \mathbb{E}\big\langle P(o_t|\Delta\ell)\big\rangle}{180°} \right| \tag{3c}$$
$$\omega^{o|o}_t = 1 - \omega^{o|\ell}_t \tag{3d}$$

where $n{=}1..N$ are uniformly-distributed samples over the range $[1°..360°]$, u_t is a measure of uniformity for $P(o_t|\cdot)$ over this range and the numerator in (3c) is forced to the range $[-180°..180°]$. u_t is in fact the *survival diagnostic* (from particle filter resampling [5, 6]). Note that $\widetilde{\omega}_t \propto 1{-}u_t$, which is 0 for the uniform distribution and 1 for the impulse distribution and is exactly the weighting behaviour desired. Equation (3c) then adjusts the weights to take into account the similarity of the two distributions' means.

Finally, the use of e_t differs from the standard HHMM (in particular, the arrow between e_t and q_t is reversed). The role of e_t in the FS-HHMM is to assist in particle-filtered inference when processing a test sequence. Apart from the usual concerns of degeneracy and good importance sampling [6], an issue with particle filtering in a hierarchical

model such as the FS-HHMM is the method of sampling particles. Specifically, for a particle (i), $P(q_t, x_t|q_{t-1}^{(i)}, x_{t-1}^{(i)})$ is sampled in its factored form $q_t^{(i)} \sim P(q_t|q_{t-1}^{(i)})$ and $x_t^{(i)} \sim P(x_t|q_t^{(i)}, x_{t-1}^{(i)})$. But it is possible that a $q_t^{(i)} \neq q_{t-1}^{(i)}$ is sampled such that $\forall x_t, P(x_t, q_t^{(i)}|x_{t-1}^{(i)}, q_{t-1}^{(i)}) = 0$. In other words, $x_{t-1}^{(i)}$ is not at a value that can transition into $q_t^{(i)}$. Moreover, since x_{t-1} is factored, *all* 24 sub-states must be at such a transition-able value, yet typically only about half of the sub-states fit this category. One solution is to add a non-zero Dirichlet prior on all x_t transitions, but this makes every x_t valid for every q_t, undermining the purpose of q_t. Another solution is to sample from $P(q_t, x_t|q_{t-1}^{(i)}, x_{t-1}^{(i)})$, but since it is rare that all 24 sub-states of x_t can move to a new phase, the particles will evolve too slowly to keep up with the true motion. Instead, an alternative sampling of x_t is allowed. If a sub-state x_{t-1}^k cannot transition into q_t, e_t^k is set to 1 and x_t^k is sampled from $P(x_t^k|q_t^{(i)})$ (*i.e.* $x_t^k \perp x_{t-1}^k$ if $e_t^k{=}1$). This uses context-specific independence on $x_t^k|e_t^k$ [2], or alternatively could be viewed as a mixture of Monte Carlo kernels for inference purposes [1]. Formally:

$$e_t^k = \begin{cases} 1 & \text{if } \forall j, P(x_t^k{=}j, q_t^{(i)}|x_{t-1}^{k^{(i)}}, q_{t-1}^{(i)}) = 0 \\ 0 & \text{otherwise} \end{cases} \quad (4a)$$

$$\text{sample } x_t^{k^{(i)}} \sim \begin{cases} A_{nij}^k \triangleq P(x_t^k|q_t^{(i)}, x_{t-1}^{k^{(i)}}, e_t^k{=}0) \\ \Lambda_{nj} \triangleq P(x_t^k|q_t^{(i)}, e_t^k{=}1) \end{cases} \quad (4b)$$

Thus e_t ensures that particles can evolve quickly whilst still being sampled from good areas of the state space. Note that during training x_t is fully observed and so e_t is always 0 (*i.e.* $x_t \rightarrow x_{t+1}$ is always valid since both are observed).

2.3. Learning and Inference

Training Data During training, the values for the hidden nodes $\{x_t, \ell_t, o_t\}$ are supplied from ground-truth data since the 29D state space is too large to explore tractably. Since x_t is discrete at 3° intervals, a smooth and accurate ground-truth for x_t is not critical. Indeed, the loose body model means that $f(y_t|x_t)$ is not sensitive enough to make full use of highly accurate ground-truth data. This raises the possibility of using a standard full-body tracker in a *controlled* environment to produce the training data instead of a marker-based motion-capture facility. This paper uses the annealed particle filter (APF) tracker with the body model and observation likelihood function from Section 2.1. To obtain clean observations, the training actor wore clothes that have good contrast between different body parts and all occluding furniture was removed. In addition, video was captured from four views at 50fps by capturing interlaced 768×576 frames at 25fps, then splitting and subsampling each frame into two 384×288 fields to realise 50fps. The resulting half-pixel vertical 'jitter' does not affect tracking. Despite these measures, the APF still can follow incorrect posture avenues due to the loose fit of the body model. Hence a human user monitors the APF's execution and manually corrects the posture whenever it begins to fail (which usually involves correcting a few frames at a time).

Learning Training is greatly simplified by observing $\{x_t, o_t, \ell_t\}$. EM is needed only to estimate parameters involving q_t using the following update equations:

$$\hat{C}_{mn} \propto \sum_{t=2}^{T} \xi_t(m,n) \quad (5a)$$

$$\hat{A}_{nij}^k \propto \sum_{t=2}^{T} \gamma_t(n) \cdot \delta(x_{t-1}^k{=}i, x_t^k{=}j) \quad (5b)$$

$$\hat{\Lambda}_{nj}^k \propto \sum_{t=2}^{T} \Big[\delta(x_t^k{=}j) \sum_{m \neq n} \xi_t(m,n)\Big] \quad (5c)$$

$$\hat{\phi}_m = \gamma_1(m) \quad (5d)$$

$$\hat{\pi}_{mi}^k \propto \gamma_1(m) \cdot \delta(x_1^k{=}i) \quad (5e)$$

where $\delta(\cdot)$ is 1 if the function's arguments are equal and 0 otherwise, and the variables $\gamma_t(m) \triangleq P(q_t{=}m|x_{1..T})$ and $\xi_t(m,n) \triangleq P(q_{t-1}{=}m, q_t{=}n|x_{1..T})$ are calculated by:

$$\gamma_t(m) = \frac{\alpha_t(m) \cdot \beta_t(m)}{\sum_h \alpha_t(h) \cdot \beta_t(h)} \quad (6a)$$

$$\xi_t(m,n) = \frac{\alpha_{t-1}(m) \cdot C_{mn} \cdot \beta_t(n) \cdot \prod_{k=1}^{24} A_{nij}^k}{\sum_h \alpha_t(h) \cdot \beta_t(h)} \quad (6b)$$

$$\alpha_t(n) = \Big(\sum_m \alpha_{t-1}(m) \cdot C_{mn}\Big) \prod_{k=1}^{24} A_{nij}^k, \quad \alpha_1(n) = \phi_n \prod_{k=1}^{24} \pi_{ni}^k \quad (6c)$$

$$\beta_t(m) = \sum_n \Big(\beta_{t+1}(n) \cdot C_{mn} \prod_{k=1}^{24} A_{nij}^k\Big), \qquad \beta_T(m) \triangleq 1 \quad (6d)$$

where $i \triangleq x_{t-1}^k$, $j \triangleq x_t^k$. See [9] for details of the derivation.

Inference and Auto-Initialisation As well as the learned model parameters, the FS-HHMM requires three additional parameters to perform inference:

(i) variance of $P(o_t|o_{t-1})$ and $P(\ell_t|\ell_{t-1})$;
(ii) estimated initial value of $\{\ell_1, o_1\}$; and
(iii) global scale of the body model.

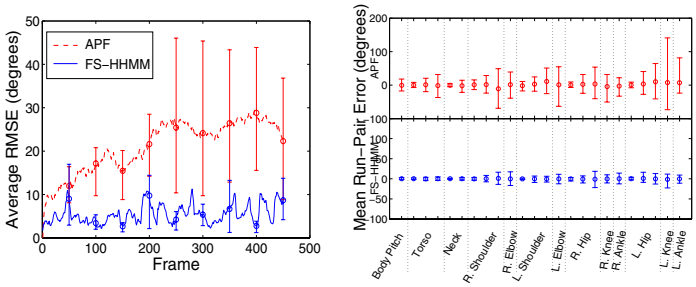
The variances for (i) are fixed and empirically set based on how fast a person is expected to move with respect to the video frame rate. The final two parameters are extracted by bootstrapping the FS-HHMM from a bounding box tracker [11]. Bootstrapping is based on the assumption that a person will walk upright into the room facing forward, thus implying scale and orientation. The system waits to bootstrap the FS-HHMM until the box-tracker reports that all four views closely agree on the person's location and height. Initial location ℓ_1 is then the box-tracker's position, initial orientation o_1 is their direction of motion and an estimate of scale is the average of each view's bounding box height. The FS-HHMM then generates candidate particles for initialisation by sampling postures from the motion model and perturbing the box-tracker and scale estimates with Gaussian noise. Particles are weighted via $f(y_t|x_t)$ and the scale is fixed at the weighted mean scale. Tracking (filtering) then proceeds with this fixed scale, which is

generally accurate to within about 5cm — good enough for the FS-HHMM to work with given that the body model is already loose-fitting.

Occlusion Processing When the person is partially occluded by scene objects (*e.g.* legs occluded by a table), it would be advantageous to limit $f(y_t|x_t)$ to evaluate the area in each view that is *not* occluded (since the occluded area provides no useable evidence). This requires detecting occlusions and determining the pixel area that is occluded; the bounding box tracker of [11] provides exactly this information. Thus the system continues to run the box-tracker even after bootstrapping the FS-HHMM, and $f(y_t|x_t)$ takes into account this occlusion data to improve its response during occlusions. This occlusion-specific processing is entirely optional, but improves tracking robustness – a failure of the box-tracker only means that $f(y_t|x_t)$ is not provided with this occlusion evidence, hence there are no negative effects beyond the small overhead of box tracking.

3. Consistency of the FS-HHMM vs the APF

This section establishes the motivation behind developing the FS-HHMM for action recognition tasks by briefly outlining the behaviour of the APF when working with a loose-fitting body model and cluttered scenes. The problem is that the APF (and indeed most full-body trackers) relies on optimising $\max_{x_t} f(y_t|x_t)$ within a search area defined by the current posture and dynamics model. This is a poor optimisation criteria when the observations y_t are noisy and error-prone. An error will mean $f(y_t|x_t)$ has multiple incorrect maxima surrounding the true posture but has no maxima at the true posture. In practice, this causes the APF to behave inconsistently, producing very different outputs depending on which incorrect maxima it chooses. A simple experiment can show this. The APF (10 layers, 1,000 particles, 10,000 total evaluations) was used to produce seven posture sequences, each 450 frames (18 seconds) long and all generated from the *same* video. The only difference between the initial conditions of the various runs was the random seed value (used for sampling), with all other factors being identical. For comparison, seven runs using the FS-HHMM (with 1,000 particles) were also conducted on the same video and starting position as the APF runs, again differing only in the random seeds. Since the runs should all be the same, the error is measured in terms of the difference between pairs of runs (21 run-pairs in total). Figure 4a plots the average over all 21 run-pairs of the root mean squared error (RMSE) of the 24 joint rotations (*i.e.* RMSE=$\frac{1}{24}\sum_{k=1}^{24}(x^k_{t,Run1} - x^k_{t,Run2})^2$). Figure 4b shows the mean and variance of each joint rotation's error across the 450 frames of all run-pairs. As can be seen the APF is much less consistent, with a higher error and greater error range than the FS-HHMM. The APF error also tends to increase with time, whereas the FS-HHMM recovers from error peaks. Note that the FS-HHMM is more consistent even though it must auto-initialise the posture based on the random seed (unlike the APF, where all runs start with exactly the same posture).



(a) (b)

Figure 4. Error between several runs of the APF and FS-HHMM on the *same* video. Runs differed *only* in their random seeding. Error bars show 5^{th} and 95^{th} percentiles. (a) RMSE of joint rotations, averaged across the 21 run-pairs. (b) Error mean and variance of each joint rotation over all frames.

Given that the APF produces such inconsistent posture sequences from a *single* video when only the random seed is varied, it would be a formidable challenge to use the APF to successfully recognise actions in *different* videos. Thus this paper does not attempt action recognition with the APF.

4. Experiments

To evaluate the FS-HHMM, four different actions were processed with both the FS-HHMM and APF. Actions are:

- *Walk* – walking, standing, and abrupt turns;
- *Sit* – sitting down onto a chair;
- *Stand* – standing up from a chair; and
- *Fridge* – getting a bottle from a bar fridge.

All *Walk* sequences are over 60 seconds in length; the other actions range between two and six seconds each. The room is monitored by four ceiling-mounted cameras, one in each corner of the room. For *Walk* the room contained one table and two chairs placed around the room, arranged in different configurations between sequences to produce different occlusion patterns. Occlusions are also caused by four fixed cupboards that stand beside each camera. The scene for *Sit*/*Stand* was arranged to include one table and four chairs, with test sequences captured by sitting in turn in each chair twice. Occlusions differed per chair, ranging from unoccluded to largely blocked by the table. For *Fridge* sequences, the fridge door itself causes occlusions when opened, and tables and chairs were placed on either or both sides of the fridge to produce various levels of occlusions for different sequences.

For the APF the user must provide the full posture, position, orientation and starting frame of each sequence. For the FS-HHMM, *Walk* requires no user input, being bootstrapped from a bounding box tracker when the person first enters the scene. However, the latter three actions cannot be bootstrapped since the person enters the scene walking, not sitting. Thus the FS-HHMM can estimate the initial pose of these actions, but requires the user to provide the starting frame and initial position. Future work will be to chain actions together so that a bootstrapped *Walk* can provide a subsequent *Sit* with its position estimate.

Four FS-HHMMs are trained, one for every action. Training data is obtained using the APF with a single, 'clean' sequence as described in Section 2.3, and is not used for testing. Each training example is 'mirrored' to account for the left-right symmetry of humans, producing a second training sequence. No other training data is used. Moreover, the training sequence for *Walk* only involves four steps along a straight path, whereas the test data includes turns, standing still and abrupt changes in direction. The minimal training data (two examples per action) is sufficient because the FS-HHMM is essentially learning a canonical example of the action, and the observation likelihood $f(y_t|x_t)$ evaluates how well the observed motion fits this canonical version. Furthermore, the assumption of conditional independence between all joint rotations $x_t^1, .., x_t^K$ means that rotation combinations are not constrained, so novel joint angles can be produced that are not in the training data.

Sit, *Stand* and *Fridge* are all modelled with 16 states and left-right transition matrices C_{mn} for q_t. *Walk* is modelled with eight states and a cyclic C_{mn} (a left-right model whose last state can transition back to the first state). State sizes were chosen empirically. A Dirichlet prior on C_{mn} is also added to allow for transitions to any later phase.

For particle-filtered inference, the FS-HHMM resamples at every time step. A resampling strategy is employed that always retains the top few particles in each phase $q_t = m$. This maintains a good spread of particles across phases, with 5% of particles retained in this way. To infer the hidden states $\{q_t, x_t, e_t, \ell_t, o_t\}$ with four views, the FS-HHMM with 1,000 particles takes approximately 12sec per frame as implemented in unoptimised C++ code running on a desktop Pentium-4 3GHz. On the same code-base the 10-layer, 1,000-particle APF (10,000 evaluations) takes about 120sec per frame, indicating that almost all processing is in evaluating $f(y_t|x_t)$ for every particle.

5. Results and Analysis

In order to test each aspect of body tracking with the FS-HHMM, several evaluations are performed that cover initialisation, tracking and failure recovery, robustness to reduced resolutions and frame rates, action recognition and tracking people other than the training actor. In terms of evaluation, *Walk* is the broadest test of tracking capability since the sequences are relatively long in duration, involve movement around the entire scene and contain numerous occlusions. In contrast, the other three actions are acyclic motions, where the person's movements are confined to a small area and there is no motion cycle to give the tracker a second chance at correcting a failure. All test sequences contain observation errors ranging from poor segmentation to significant occlusions. Videos of some of the results are also available.[1]

Initialisation Automatic initialisation of the FS-HHMM is an implicit part of all of the tests in this section. To test its behaviour explicitly, experiments were conducted to evaluate the *time-to-lock-on* (*i.e.* the number of frames it takes for the FS-HHMM to attain an accurate posture after initialisation). To this end, the ground-truth initial positions $(x, y, z, orientation)$ of four *Walk* postures are randomly perturbed by ($\pm$0.5m, $\pm$0.5m, $\pm$0.1m, $\pm30°$), with 12 perturbations of each posture (48 in total). The four postures are in different phases of the *Walk* motion and at different locations in the room. *Walk* is tested since its range of starting postures provides the most challenge to initialisation.

The resulting distribution of the time-to-lock-on is roughly exponential with a mean of 6.77 frames (just over a quarter of a second given the 25fps video). No perturbation took more than 20 frames to achieve an accurate lock.

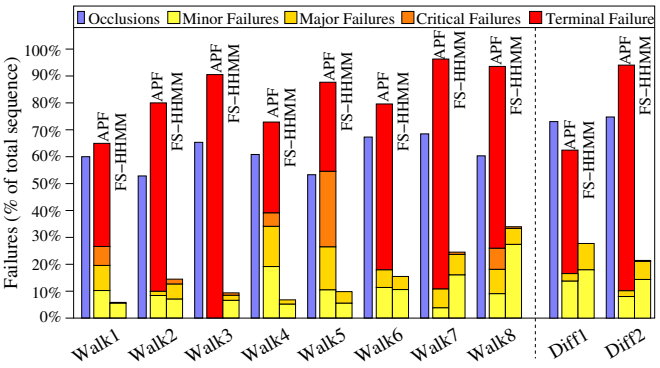


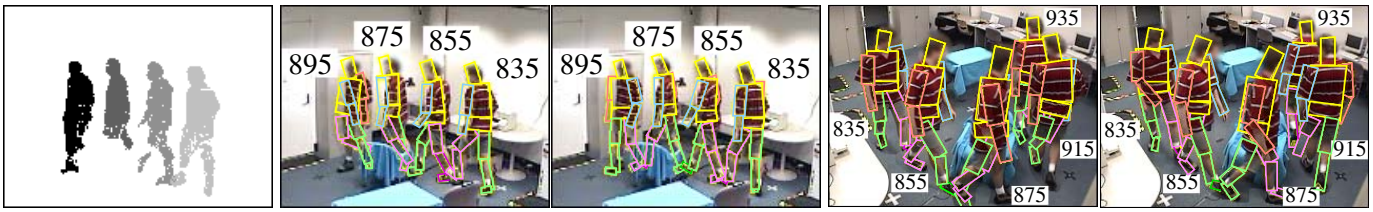
Figure 6. Failures for *Walk* sequences for both the APF and FS-HHMM. The narrow blue bars indicate the proportion of frames in which one or more views are occluded.

Walk Tracking In lieu of a ground-truth for the posture sequences, the APF and FS-HHMM are analysed in terms of manually identifying and categorising noticeable tracking failures according to their severity. The proportion of frames spent in each failure category thus indicates the relative robustness of the APF and FS-HHMM to observation errors. There are four failure categories:

- *Minor:* Legs locked together, poor body orientation;
- *Major:* Swapped legs, contorted postures;
- *Critical:* Reversed body orientation, leg folded over;
- *Terminal:* Major/critical failure that is never corrected.

Failures such as slight limb inaccuracies are not included. Figure 6 shows the failures as a proportion of the total duration for each *Walk* sequence, including *Walk* sequences of two different people (*Diff 1* and *Diff 2*) who are respectively shorter and taller than the training actor. Note that the FS-HHMM suffers fewer failures and never fails terminally. Failures also persist for a shorter duration due to the self-correcting mechanisms of the model. This is despite the fact that the FS-HHMM receives no user input – position, orientation and scale are all initialised by bootstrapping from a box-tracker. In contrast, whenever an occlusion occurs the APF usually fails. Worse, the APF has terminal failure in *every* sequence and cannot recover due to its local search approach. Note that performance does not alter much when tracking the *Diff* walkers, showing the robustness of

[1] `www.cs.curtin.edu.au/~peursump/demos.html`



Background - NE View | APF - NE View | FS-HHMM - NE View | APF - SW View | FS-HHMM - SW View

Figure 5. Example time lapse sequence of two views covering 100 frames and comparing the APF and FS-HHMM. Although both start with good postures, the APF fails terminally (from body reversal) by frame 895 whereas the FS-HHMM only experiences a minor failure.

bootstrapping and tracking to different people.

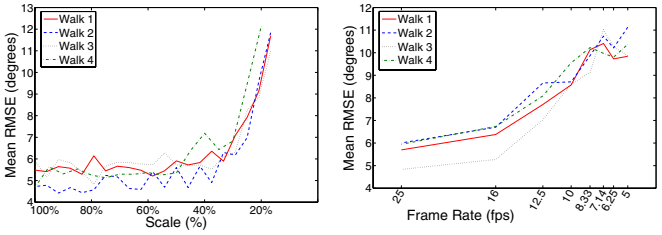
The frequent observation errors means that the APF is rarely able to maintain good-quality tracking for more than a few frames without producing trivial failures or unnatural movements. On the other hand, when observing conditions are persistently clean the APF is often more accurate than the FS-HHMM since it is not constrained by any motion model. Surprisingly however, the FS-HHMM does not degrade noticeably when tracking motions that moderately diverge from the training data. Specifically, standing still or leaning into a turn are tracked well despite not being in the training data. However, walking backwards or non-walking motions will cause the FS-HHMM to fail.

	APF			FS-HHMM
Action →	Sit	Stand	Fridge	All
# Occluded Seqs	4 of 8	4 of 8	8 of 8	(see left)
Fail Count	3 (38%)	4 (50%)	2 (25%)	0 (0%)
Mean TTF	0.53s	0.72s	1.82s	-
TTF Range	0.4s-0.72s	0.52s-1.16s	1.08s-2.56s	-

Table 1. Terminal failure counts mean time-to-failure (TTF) and TTF range (min to max) for each action. Test videos are at 25fps.

Sit, Stand, Fridge Tracking These actions are analysed differently to *Walk* since they are very short and so any failure is invariably terminal. Instead, they are evaluated by considering the number of sequences that fail terminally and the mean time until this failure – see Table 1. The APF tracks very accurately in sequences with no occlusions since the three actions involve fairly sedate motions, but fails terminally in most sequences (9 of 12) that contain occlusions (in *Walk*, every sequence contained multiple occlusions). As with *Walk*, the FS-HHMM is much less sensitive to occlusions and never fails terminally. However, the APF is definitely more accurate when observations are clean since the FS-HHMM is constrained by its motion model. The FS-HHMM can also lag slightly behind some motions – for example, during *Stand* the shadow on the chair seat can cause the FS-HHMM to be temporarily stuck in half-seated posture (see third *Stand* frame of Figure 8). This is corrected a few frames later as the person continues to rise, hence smoothing [6] may help to minimise this issue.

Although the *Fridge* action has fewer terminal failures for the APF and none for the FS-HHMM, both the FS-HHMM and the APF are not very accurate in tracking the person's arms during *Fridge* since the opened door is detected as foreground and obscures the arms (see Figure 8). This causes the two trackers to produce multiple minor failures in arm tracking which are corrected only when the person closes the door. In addition, the motions for executing *Fridge* are not well-constrained by the action's purpose – how the fridge door is opened (*e.g.* from the front or side, near or far from the fridge) is less important than the fact that it gets opened. Hence the motion model of the FS-HHMM is not as beneficial as it is for the other actions.



(a) (b)

Figure 7. Mean error of several video sequences as their resolution (a) and frame rate (b) are reduced (all other factors are constant).

Low Resolution, Low Frame Rates Figure 7 plots the performance of the FS-HHMM when resolution and frame rates are reduced on several different videos of *Walk*. The graphs plot the difference (mean RMSE) between each subsampled case and the base case of 384×288 and 50fps. Resolution was reduced using bicubic subsampling (the 100% scale case is a 384×288 bicubic *re*-sampling). Note that a relatively low RMSE is around 5 or less (see Figure 4a). Thus Figure 7a shows that the FS-HHMM is suprisingly robust to decreasing resolution – the error remains low until the resolution is more than halved (about 160×120). This robustness is likely due to the fact that the observations corroborate the motion model rather than the other way around, Hence lower detail can still provide useable evidence, especially given the distinctive phases of the walking cycle. Figure 7b shows a steady, roughly exponential, degradation when lowering the frame rate. Significant inconsistency with the base 50fps case begins to show at frame rates below 16.7fps, and *Walk3* is still quite consistent even at 12.5fps. This ability to handle reduced frame rates arises from the e_t node, which allows particles to 'skip ahead' to new phases without 'finishing' the current phase.

Action Recognition Action recognition is evaluated by running each model (*Walk*, *Sit*, *Stand*, and *Fridge*) against all 34 sequences (8 per action, plus the two *Diff* walking sequences). The approximate log-likelihood [6] of each run is then calculated and the most likely model is assigned as

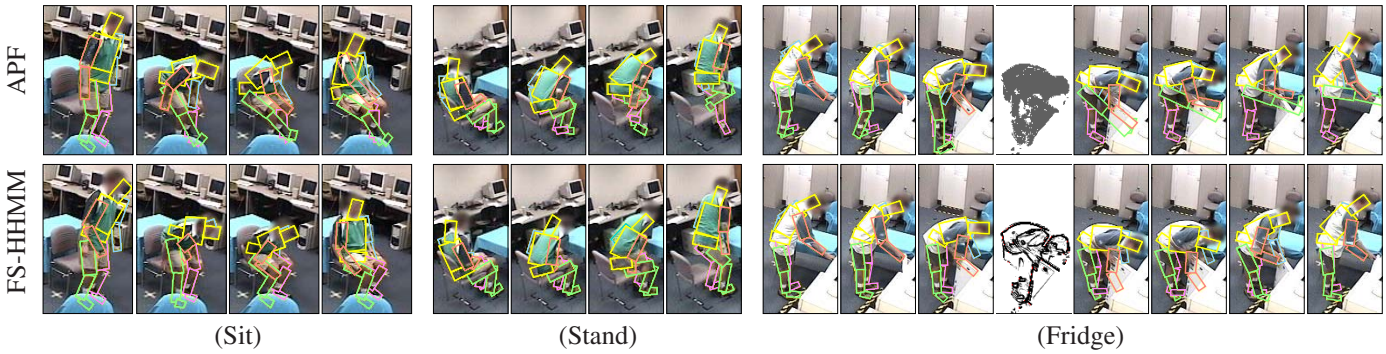


Figure 8. Examples sequences of *Sit*, *Stand* and *Fridge* showing every 10^{th} frame. Although the APF tracks more accurately at first, the occlusions (*Sit*, *Stand*) and foreground errors (*Fridge*) cause it to fail terminally, whereas the FS-HHMM does not. Note that in *Fridge* both the APF and FS-HHMM have difficulty tracking the arms when the fridge door is opened since the door becomes part of the foreground.

the action label. Table 2 shows the results. Although the list of actions is small, the 100% accuracy of recognition is encouraging given the high-dimensionality of the features and the similarity of motions like *Sit* and *Fridge*. In contrast, the noise and inconsistency of the APF (Section 3) makes learning distinctive models of different motions based on the APF's outputs a formidable challenge, hence action recognition with the APF was not attempted.

Classed as →	Walk	Sit	Stand	Fridge	Recall
Walk	10	0	0	0	100%
Sit	0	8	0	0	100%
Stand	0	0	8	0	100%
Fridge	0	0	0	8	100%
Precision	100%	100%	100%	100%	100%

Table 2. Confusion matrix for action recognition.

6. Conclusion

This paper presents a model (the FS-HHMM) that fuses human body tracking with action recognition, and shows that this can significantly improve robustness to observation errors such as occlusions, poor segmentation and reduced resolution. The approach also facilitates auto-initialisation and self-corrections when failures do occur. A range of experiments were conducted to test all of these aspects. Comparison with the APF showed that although the FS-HHMM's tracking is not as accurate as the APF when viewing clean observations and sedate motion, when conditions are not so favourable the APF quickly fails and frequently cannot recover, whereas the FS-HHMM is far less fragile.

Issues that still face the FS-HHMM include testing the system with a greater variety of actions and the need to unite the individual action models into a larger hierarchy so that actions can be chained together and segmented automatically. Unmodelled motions could be then represented by an 'action' that is based on a generic motion model similar to the APF. Finally, accurately tracking actions whose motions can vary widely must also be addressed, as shown by the difficulty of tracking the *Fridge* action.

Acknowledgements This research is supported by Australian Research Council grant LP0561867. We would also like to thank iVEC (`www.ivec.org`) for the use of their computer facilities.

References

[1] C. Andrieu, N. De Freitas, A. Doucet, and M. I. Jordan. An introduction to MCMC for machine learning. *Machine Learning*, 50:5–43, 2003.

[2] C. Boutilier, N. Friedman, M. Goldszmidt, and D. Koller. Context-specific independence in Bayesian networks. In *Uncertainty in AI*, pages 115–123, 1996.

[3] H. Bui, D. Phung, and S. Venkatesh. Hierarchical hidden Markov models with general state hierarchy. In *AAAI 2004*, pages 324–329, 2004.

[4] F. Caillette, A. Galata, and T. Howard. Real-time 3-D human body tracking using Variable Length Markov Models. In *British Machine Vision Conference*, pages 469–478, 2005.

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

[6] A. Doucet, S. Godsill, and C. Andrieu. On sequential Monte-Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10(3):197–208, 2000.

[7] M. W. Lee, I. Cohen, and S. K. Jung. Particle filter with analytical inference for human body tracking. In *IEEE Workshop on Motion and Video Computing*, pages 159–165, 2002.

[8] I. Mikíc, M. Trivedi, E. Hunter, and P. Cosman. Articulated body posture estimation from multi-camera voxel data. In *IEEE CVPR*, volume I, pages 455–460, 2001.

[9] P. Peursum. A factored-state HHMM for articulated human motion modelling. Technical report, Curtin University of Technology, 2006. impca.cs.curtin.edu.au/pubs/reports.php.

[10] P. Peursum. On the behaviour of the annealed particle filter in realistic conditions. Technical report, Curtin University of Technology, 2006. impca.cs.curtin.edu.au/pubs/reports.php.

[11] P. Peursum, S. Venkatesh, and G. West. Observation-switching linear dynamic systems for tracking humans through unexpected partial occlusions by scene objects. In *Int'l Conf. on Pattern Recognition*, pages IV:929–934, 2006.

[12] H. Sidenbladh, M. J. Black, and D. J. Fleet. Stochastic tracking of 3D human figures using 2D image motion. In *European Conference on Computer Vision*, pages 702–718, 2000.

[13] C. Sminchisescu and A. Jepson. Variational mixture smoothing for non-linear dynamical systems. In *IEEE CVPR*, volume 2, pages 608–615, 2004.

[14] T. Zhao and R. Nevatia. 3D tracking of human locomotion: A tracking as recognition approach. In *Int'l Conf. on Pattern Recognition*, volume 1, pages 546–551, 2002.