Markerless Motion Capture of Complex Full-Body Movement for Character Animation

Andrew J. Davison, Jonathan Deutscher and Ian D. Reid
Robotics Research Group
Department of Engineering Science
University of Oxford
Oxford OX1 3PJ, UK
[ajd,jdeutsch,ian]@robots.ox.ac.uk

Abstract. Vision-based full-body tracking aims to reproduce the performance of current commercial marker-based motion capture methods in a system which can be run using conventional cameras and without the use of special apparel or other equipment, improving usability in existing application domains and opening up new possibilities since the methods can be applied to image sequences acquired from any source. We present results from a system able to perform robust visual tracking with an articulated body model, using data from multiple cameras. Our approach to searching through the high-dimensional model configuration space is an algorithm called *annealed particle filtering* which finds the best fit to image data via multiple-layer propagation of a stochastic particle set. This algorithm efficiently searches the configuration space without the need for restrictive dynamical models, permitting tracking of agile, varied movement. The data acquired can readily be applied to the animation of CG characters. Movie files illustrating the results in this paper may be obtained from **http://www.robots.ox.ac.uk/~ajd/HMC/**

1 Introduction

Motion capture aids animators in the time-consuming task of making computer graphics characters move in realistic ways by enabling movement data to be recorded straightforwardly from observation of an actor performing the desired motions. Commercial human motion capture technology [16] has now been used for CG character animation for a number of years. In typical current systems, a number of retro-reflective markers are attached in known positions to the actor's body and viewed by infra-red cameras emitting light and filtered to remove background signals. The easily-recovered image positions of the markers are transformed into 3D trajectories via triangulation of the measurements from multiple cameras, and a parameterised representation of the actor's movements can be calculated.

It is a sign of the growing synergy between graphics and computer vision (seen recently in areas such as augmented reality [15]) that visual processing can in fact be used to recover motion data *directly from images*, without markers, and many analogues can be seen in the processes used for tracking and those used later on to reanimate the data in graphical output. The use of markers is intrusive and restricting, necessitates the use of expensive specialised capture hardware, and requires footage to be taken specially. The goal of markerless motion capture is to reproduce the performance of marker-based methods in a system which could be run using conventional cameras and without the use of special apparel or other equipment.

Such a system would of course be able to replace current marker-based systems, improving usability in studio scenarios, but also potentially be used in a variety of new domains such as sports broadcasting. However, full-body tracking from standard images is a challenging problem, and research has so far failed to produce a full-body tracker general enough to handle real-world applications. In particular, no markerless system presented to date has convincingly achieved the following combination of capabilities of current marker-based systems which would make it a viable alternative in animation applications: full 3D motion recovery; robust tracking of rapid, arbitrary movement; high accuracy; easy application to new scenarios.

A number of effective systems able to recover 2D motion from a single camera have been presented [6, 10]. While these might be useful in simple applications, tracking "side-on" planar movements, they do not provide output in the form of 3D model configurations that are needed in general for character animation.

Bregler and Malik [2] produced some of the best-known results to date in 2D and 3D body tracking. Their approach was based on frame-to-frame region-based matching using a gradient-descent search and was demonstrated on several short multi-camera image sequences. However, this simple search scheme is not capable of tracking agile motions with cluttered backgrounds, and their method of locating body parts by frame-to-frame matching of image regions will cause drift over long sequences. Howe *et al.* [8] present a system for single-camera tracking which combines a 2D tracker with learned models of 3D configuration to produce 3D pose output for simple sequences.

Gavrila and Davis [5] use an explicit hierarchical search, in which parts of the body's kinematic chain are located sequentially (e.g. torso, followed by upper arm, lower arm and then hand), a process which greatly reduces search complexity. Without the assistance of artificial labelling cues (such as colour), however, it is very hard to localise specific body parts independently in realistic scenarios. This is due to the fact that limited measurements of a specific body part itself may not be sufficient to recover its own position: information on the location of parts further down the kinematic hierarchy also give vital information (for instance the orientation of the torso may not be apparent until arms are observed).

In work with many similarities to that in this paper, Sidenbladh *et al.* [14] have taken a mathematically rigorous approach to full-body tracking based on Condensation (see Section 2.1), using learned dynamical models and a generative model of image formation. They tracked short sequences of 3D motion from a single camera, though the very strong dynamical models used restrict the applicability of the system to general motion tracking and the system runs slowly due to the large number of particles required.

Building on previous work in [4], in this paper we present a method for full body tracking using multiple cameras. Our approach is characterised by the following: 1. articulated body model, 2. weak dynamical modelling, 3. edge and background subtraction image measurements, and 4. a particle-based stochastic search algorithm. The latter uses a continuation principle, based on annealing, to introduce the influence of narrow peaks in the fitness function gradually. The algorithm, termed *annealed particle filtering*, is shown to be capable of recovering full articulated body motion efficiently, and demonstrated tracking extended walking, turning and running sequences.

We will introduce and review visual tracking in general in Section 2, then move on to a discussion of the specifics of full-body tracking in Section 3. The annealed particle filtering approach which this problem has led us to take is described in Section 4. Section 5 presents results from tracking agile walking, running and handstand movments.

2 Visual Tracking

Full-body motion capture is an example of *model-based tracking*, in that it is the process of sequentially estimating the parameters of a simplified model of a human body over time from visual data. The parameters $\mathbf{X}$ needed to specify a particular state of the model are called its degrees of freedom. As well as models representing the *shape* and *motion* of the target, a model of the *measurement* process, through which information is gained from images in the form of measurement parameters $\mathbf{Z}$, is required.

In early model-based tracking using vision (e.g. [11]), the targets were simple objects which could closely be modelled with mathematically convenient geometrical shapes, and clear edges or other image features were available as reliable measurements. In cases like these, localising the object at each time step can proceed as a gradient-descent search in which a measure of fit of a hypothesized model configuration is repeatedly evaluated based on how well it predicts the measurements obtained with the model degrees of freedom deterministically adjusted to find the global best fit. Bregler and Malik [2] used a similar method in their work on full-body tracking.

An unconstrained search of this type, generally initialised at each new time step at the model configuration found in the previous frame, can get into trouble with local maxima in the search space, and will not be able to track rapid movement. It is profitable to constrain the search area using information which is available about the possible motion of the object: given knowledge about where the object was at a sequence of earlier time steps, it is possible to make a prediction, with associated uncertainty, about where it will be at the current time, and limit search to this part of configuration space.

When combining motion and measurement information in this way however, we can do better than simply using motion information to initialise a search by putting both types of information into an absolute, Bayesian probabilistic framework. The Extended Kalman Filter (EKF) has been widely used in visual tracking [7] to achieve this. The "goodness of fit" function associated with measurements must now take the form of a likelihood $p(\mathbf{Z}|\mathbf{X})$ which describes the probability of measurements $\mathbf{Z}$ given a state $\mathbf{X}$, and the motion model has the form $p(\mathbf{X}_k|\mathbf{X}_{k-1})$. Tracking now proceeds as a sequential propagation of a probability density function in configuration space: the estimate of model configuration at any time step is a weighted combination of both information from the most recent set of measurements and, via motion continuity, that from previous measurements.

In more difficult tracking problems, where the models were now for example deformable 2D templates tracking complicated objects with agile motion, EKF-based tracking was enhanced with the use of learned motion models [13]: analysis of a training data set enabled probabilistic models of motion to be built, giving much better tracking of future motions of the same type. Baumberg and Hogg applied methods of this kind to the tracking of human figures from a single camera [1], obtaining good estimates of global movement but not the details of articulation needed for motion capture.

2.1 Particle Filters

The EKF provides a probabilistic framework for tracking, but supports only the case where observation and motion probability density functions can be approximated as multi-variate Gaussians. While Gaussian uncertainty is sufficient for modelling many motion and measurement noise sources, the EKF has been shown to fail catastrophically in cases where the true probability function has a very different shape. Attempts to track objects moving against a very cluttered background, where measurement densities include the chance of detecting erroneous image features, led to the first application of particle filtering in visual tracking [9] in the form of the Condensation algorithm.

In particle filtering, the posterior density $p(\mathbf{X}|\mathbf{Z}_k)$ representing current knowledge about the model state after incorporation of all measurements is represented by a finite set of *weighted particles*, or samples, $\{(\mathbf{s}_k^{(0)}, \pi_k^{(0)}) \ldots (\mathbf{s}_k^{(N)}, \pi_k^{(N)})\}$ where the weights $\pi_k^{(n)} \propto p(\mathbf{Z_k}|\mathbf{X} = \mathbf{s}_k^{(n)})$ are normalised so that $\sum_N \pi_k^{(n)} = 1$. The state $\mathcal{X}_k$ at each time step t_k can be estimated by the sample mean:

$$\mathcal{X}_k = \mathcal{E}_k[\mathbf{X}] = \sum_{n=1}^{N} \pi_k^{(n)} \mathbf{s}_k^{(n)} \tag{1}$$

or the mode

$$\mathcal{X}_k = \mathcal{M}_k[\mathbf{X}] = \mathbf{s}_k^{(j)}, \pi_k^{(j)} = \max(\pi_k^{(n)}) \tag{2}$$

of the posterior density $p(\mathbf{X}|\mathbf{Z}_k)$. Variance and other high-order moments of the particle set can also easily be calculated.

Essentially, a smooth probability density function is approximated by a finite collection of weighted sample points, and it can be shown that as the number of points tends to infinity the behaviour of the particle set is indistinguishable from that of the smooth function. Tracking with a particle filter works by: 1. Resampling, in which a weighted particle set is transformed into a set of evenly weighted particles distributed with concentration dependent on probability density; 2. Stochastic movement and dispersion of the particle set in accordance with a motion model to represent the growth of uncertainty during movement of the tracked object; 3. Measurement, in which the likelihood function is evaluated at each particle site, producing a new weight for each particle proportional to how well it fits image data. The weighted particle set produced represents the new probability density after movement and measurement.

Particle filtering works well for tracking in clutter because it can represent arbitrary functional shapes and propagate multiple hypotheses. Less likely model configurations will not be thrown away immediately but given a chance to prove themselves later on, resulting in more robust tracking.

The complicated nature of the observation process during human motion capture causes the posterior density to be non-Gaussian and multi-modal as shown experientally by Deutscher *et al.* [3], and Condensation has been implemented successfully for short human motion capture sequences by Sidenbladh *et al.* [14]. However, serious problems arise with Condensation in the high-dimensional configuration spaces occurring in human motion capture and other domains: essentially, the number of particles needed to populate a high-dimensional space is far too high to be manageable. We will explain the specifics of the full-body tracking problem in the following section, then present our approach to efficient particle filtering in Section 4.

3 Models for Full-Body Tracking

3.1 Kinematics and Dynamics

In common with the majority of full-body tracking approaches, we have used an articulated model in which the body is approximated as a collection of rigid segments joined by rotating joints. Degrees of freedom (DOF) in the model are close approximations to the the way the human skeleton moves. In joints (such as the shoulder) which have

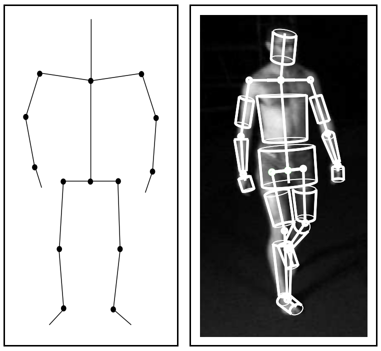


Fig. 1. (a) Typical kinematic model with 33 degrees of freedom based on a kinematic chain consisting of 18 segments. Six degrees of freedom are given to base translation and rotation. The shoulder and hip joints are treated as sockets with 3 degrees of freedom, the clavicle joints are given 2 degrees of freedom and the neck, elbows, wrists, hips, knees and ankles are modelled as hinges requiring only one. This results in a configuration vector $\mathbf{X} = \{x_1 \ldots x_{33}\}$. The model is fleshed out by conical sections (b).

more than one DOF, rotations are parameterised with sequential rotations about perpendicular axes (simlilar to Euler Angles). A reasonable articulated model of the human body usually has at least 25 DOF; we have most commonly used a model with 33 DOF (see Figure 1). Models for commercial character animation usually have over 40 DOF.

In our system, kinematic models are specified in a format commonly used in CG packages, meaning that they are easily reconfigurable and the data acquired readily applied to character animation. In addition to defining degrees of freedom, we specify range limits for each joint, hard constraints beyond which movement is not allowed. Depending on the application, decisions are made about exactly how closely to model the skeleton: commonly, it will be more important to recover gross body motion robustly than to capture in detail the positions of hands and feet for instance. In general, stable tracking becomes more difficult as the number of degrees of freedom in the model increases, since a model with high dimensionality is "looser" and more easily distracted by ambiguous image data. There is a strong argument in fact for using *different* models for tracking and animation, since finely-detailed movements (of the hands and feet for example) can potentially be added to an animation at a later stage. In the long-term, it will be desirable to refine a tracking model which can be used in many different situations: progress towards this can be seen in the slightly more advanced model used to track the handstand sequence of Section 5.

While we have taken care with our *kinematic* model to represent possible body movements realistically, a much looser approach has been taken to *dynamic* modelling of the way that the position of the body at one time step affects the position at the next: we have either dispensed with a motion model altogether, initialising the search for each time step from the model configuration found previously, or used a very simple damped velocity model for the motion at each joint (although our method does not preclude the use of an arbitrarily complex motion model).

Specifically, we have chosen not to used trained dynamical models [13, 14]. While these models aid robust tracking greatly, they restrict the motions which can be tracked to be similar to those observed in the training set. In character animation, the value of motion capture is to be able to supply interesting and unusual movements automatically, and these movements will often not be in a training set. In addition, a strong dynamical model is time-consuming to train, and a requirement of a workable motion capture system is the ability to apply it quickly in new situations. Depending on the application, there may be a desire to change the details of the model — for instance the number of degrees of freedom, or perhaps to use only a partial body model. It would be impractical to provide pre-trained models for each of these situations.

3.2 Appearance Modelling and Image Measurement

Orthogonal to the specification of kinematic and dynamic models is the choice of method used to evaluate how well hypothesized model configurations agree with image data: the role of markers in current commercial motion capture systems must be replaced by repeatable image-based measurement. In a particle filter framework, each particle representing a hypothesized model configuration must be assigned a *weight* (technically a likelihood) representing its fit to current image data.

Unlike some recent authors [14], we use a tracking model which does not aim to be *generative* in the sense that can be used to render realistic images of a person. Such a model, while desirable from the Bayesian point of view, requires a form of texture mapping, and potentially complicated effects such as lighting conditions to be taken into account, making it highly specific to a particular person, set of clothing and conditions. Our image measurement strategy was chosen considering the following criteria:

- *Generality.* The image features used should be invariant under a wide range of conditions so that the same tracking framework will function well in a broad variety of situations.
- *Simplicity.* In an effort to make the tracker as efficient as possible the features used must be easy to extract.

Two image types of image feature were chosen to construct a weighting function: 1. *edges* and 2. *foreground segmentation*. From a particular hypothesized model configuration, the locations in images at which these features are expected to appear can be predicted: i.e. edges at the boundaries of body parts, non-background regions at any positions covered by body parts. A test can then be made against the actual edges and foreground regions found in the images from bottom-up image processing to evaluate this configuration and assign it a weight. The process is described in detail in Figure 2.

Sum-of-squared difference (SSD) measures $\Sigma^e(\mathbf{X},\mathbf{Z})$ and $\Sigma^r(\mathbf{X},\mathbf{Z})$ are computed for the edge and foreground measurements respectively: these functions represent the degree of fit between the hypothesized model configuration and edge and foreground measurements with single numbers (in both cases the 0 corresponds to a perfect fit):

$$\Sigma^e(\mathbf{X},\mathbf{Z}) = \frac{1}{N^e}\sum_{i=1}^{N^e}\left(1 - p_i^e(\mathbf{X},\mathbf{Z})\right)^2 \quad ; \quad \Sigma^r(\mathbf{X},\mathbf{Z}) = \frac{1}{N^r}\sum_{i=1}^{N^r}\left(1 - p_i^r(\mathbf{X},\mathbf{Z})\right)^2 . \quad (3)$$

To combine the edge and region measurements the two SSD's are added together and the result exponentiated to give:

$$w(\mathbf{X},\mathbf{Z}) = \exp - \left(\Sigma^e(\mathbf{X},\mathbf{Z}) + \Sigma^r(\mathbf{X},\mathbf{Z})\right) . \quad (4)$$

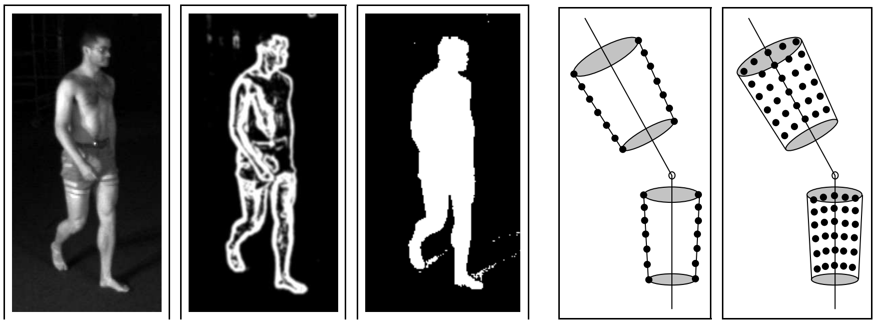


Fig. 2. Image measurement. Starting with input image (a), **two** types of bottom-up image processing are applied: (b) an edge detection mask is used to find edges, which are thresholded and smoothed using a Gaussian mask to produce a pixel map in which the value of each pixel ranges from 0 to 1 according to its proximity to an edge; (c) the foreground is segmented using thresholded background subtraction (a reference image with no target present is subtracted pixel-by-pixel from the current image) to produce a pixel map in which the value 1 correponds to foreground regions and 0 to background. These two measured pixel maps are then sampled at sites determined by each hypothesized model configuration: (d), for the edge pixel map, at sites lying along the occluding contours of the model's cone segments, providing N^e measurements with values p_i^e, and (e), for the foreground segmentation map, at sites evenly spread across the image regions spanned by the cones, providing N^r measurements with values p_i^r.

The function is trivially extended to simultaneous measurements from multiple cameras: the SSD's from each camera are simply added together:

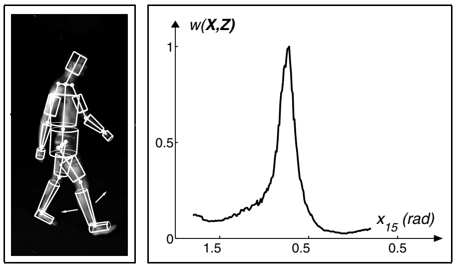
$$w(\mathbf{X}, \mathbf{Z}) = \exp - \left(\sum_{i=1}^{C} \left(\Sigma_i^e(\mathbf{X}, \mathbf{Z}) + \Sigma_i^r(\mathbf{X}, \mathbf{Z}) \right) \right) \tag{5}$$

where C is the number of cameras and $\sum_i^*(\mathbf{X})$ is from camera i. An example of the output of this weighting function, demonstrating its ability to differentiate between hypothesized configurations around a good match, can be seen in Figure 3.

Both of these measurement types can be expected to work in general imaging conditions, though clearly their performance depends on the characteristics of the particular images: neither would perform well if the target person was of an intensity profile similar to the background and therefore poorly distinguished. Foreground segmentation of the type used here of course relies on the cameras being stationary and there being little background motion. Edge measurements may fail if the person is wearing loose clothing which moves significantly relative the the rigid structure of the skeleton.

4 A Particle Filter for High-Dimensional Spaces

Particle filters such as Condensation permit robust tracking by representing arbitrary probability density functions and propagating multiple hypotheses, but a price is paid for these attributes in computational cost. The most expensive operation in the standard Condensation algorithm is an evaluation of the likelihood function $p(\mathbf{Z_k}|\mathbf{X} = \mathbf{s}_k^{(n)})$ and this has to be done once at every time step for every particle. To maintain a fair



(a) (b)

Fig. 3. Example output of the weighting function obtained by varying only component x_{15} of **X** (the right knee angle) using the image and model configuration seen in (a). The function is highly peaked around the correct angle of -0.7 radians (b).

representation of $p(\mathbf{X}|\mathbf{Z_k})$ a certain number of particles are required, and this number grows with the size of the model's configuration space. In fact it has been shown by MacCormick and Blake [12] that

$$N \geq \frac{\mathcal{D}_{min}}{\alpha^d} , \tag{6}$$

where N is the number of particles required and d is the number of dimensions. The survival diagnostic $\mathcal{D}_{min}$ and the particle survival rate α are both constants with $\alpha <<$ 1. When d is large, as in full-body tracking, normal particle filtering becomes infeasible.

Given this critical factor, combined with the fact the we have already moved away from a purely Bayesian framework in our choice of simple and generic measurement processes, it was decided to reduce the problem from propagating the conditional density $p(\mathbf{X}|\mathbf{Z_k})$ using $p(\mathbf{Z}|\mathbf{X})$ to finding the configuration $\mathcal{X}_k$ which returns the maximum value from a simple and efficient weighting function $w(\mathbf{Z_k}, \mathbf{X})$ at each time t_k, given $\mathcal{X}_{k-1}$. By doing this gains will be made on two fronts. It should be possible to make do with fewer likelihood (or weighting function) evaluations because the function $p(\mathbf{X}|\mathbf{Z_k})$ no longer has to be fully represented, and an evaluation of a simple weighting function $w(\mathbf{Z_k}, \mathbf{X})$ should require minimal computational effort when compared to an evaluation of the observation model $p(\mathbf{Z_k}|\mathbf{X})$.

We continue to use a particle-based stochastic framework because of its ability to handle multi-modal likelihoods during the search process, or in the case of a weighting function, one with many local maxima. The question is: *What is an efficient way to perform a particle based stochastic search for the global maximum of a weighting function with many local maxima?* We use a solution derived from simulated annealing.

4.1 Annealed Particle Filtering

Simulated annealing is a well known procedure in optimisation for finding the global maximum of a function which has multiple local peaks. Taking its name from the sequential coarse-to-fine adjustment of temperature needed to remove imperfections from solid structures in physics, it is the process of searching for functional maxima by first searching coarsely over a wide area, aiming to locate the approximate position in model space of the global maximum and avoid getting trapped by local maxima, and then perturbing the results achieved by smaller and smaller amounts until the global maximum can be settled on with accuracy. The amount of perturbation applied at each step, or *layer*, is the analogue of temperature in physics.

The annealed particle filter, explained in detail in [4], has the following key features which differentiate it from Condensation:

1. A layered search, in which a particle set is resampled, dispersed and weighted depending on measurements multiple times for each tracking time-step. The amount of dispersal, implemented as stochastic movement of the particle set, decreases layer-by-layer.
2. The output of the search is no longer a particle set which meaningfully represents a probability distribution, but a highly clustered group indicating the global maximum of the search.
3. The use of noise functions in the dispersion step which no longer represent purely the uncertainty associated with motion: since only a single peak of the distribution is being maintained, this additive noise must also take into account the possibility that the current estimate is substantially wrong if it is to be possible to recover from temporary tracking failures.

1. For every time step t_k an annealing run is started at layer M, with $m = M$.
2. Each layer of an annealing run is initialised by a set of un-weighted particles $\mathcal{S}_{k,m}$.
3. Each of these particles is then assigned a weight $\pi^{(i)}_{k,m} \propto w_m(\mathbf{Z}_k, \mathbf{s}^{(i)}_{k,m})$ which are normalised so that $\sum_N \pi^{(i)}_{k,m} = 1$. The set of weighted particles $\mathcal{S}^{\pi}_{k,m}$ has now been formed.
4. N particles are drawn randomly from $\mathcal{S}^{\pi}_{k,m}$ with replacement and with a probability equal to their weighting $\pi^{(i)}_{k,m}$. As the n^{th} particle $\mathbf{s}^{(n)}_{k,m}$ is chosen it is used to produce the particle $\mathbf{s}^{(n)}_{k,m-1}$ using $\mathbf{s}^{(n)}_{k,m-1} = \mathbf{s}^{(n)}_{k,m} + \mathbf{B}_m$ where $\mathbf{B}_m$ is a multi-variate gaussian random variable with variance $\mathbf{P}_m$ and mean $\mathbf{0}$.
5. The set $\mathcal{S}_{k,m-1}$ has now been produced which can be used to initialise layer $m-1$. The process is repeated until we arrive at the set $\mathcal{S}^{\pi}_{k,0}$.
6. $\mathcal{S}^{\pi}_{k,0}$ is used to estimate the optimal model configuration $\mathcal{X}_k$ using $\mathcal{X}_k = \sum_{i=1}^{N} \mathbf{s}^{(i)}_{k,0} \pi^{(i)}_{k,0}$.
7. The set $\mathcal{S}_{k+1,M}$ is then produced from $\mathcal{S}^{\pi}_{k,0}$ using $\mathbf{s}^{(n)}_{k+1,M} = \mathbf{s}^{(n)}_{k,0} + \mathbf{B}_0$. This set is then used to initialise layer M of the next annealing run at t_{k+1}.

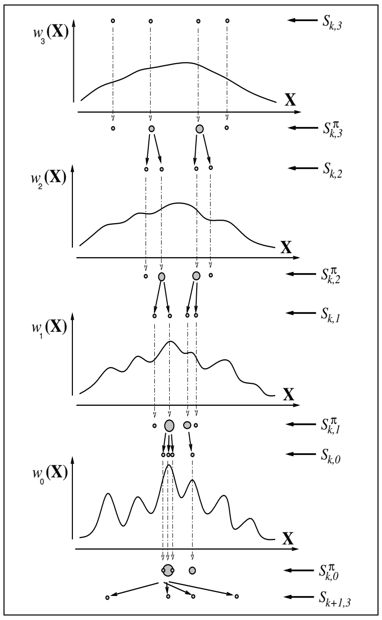


Fig. 4. Annealed particle filter algorithm and graphical depiction. With a multi-layered search the sparse particle set is able gradually to migrate towards the global maximum without being distracted by local maxima. The final set $\mathcal{S}^{\pi}_{k,0}$ provides a good indication of the weighting function's global maximum.

The annealed particle filter algorithm is given in Figure 4, along with an illustration of its action in a one-dimensional diagram which can be thought of as a "slice" through multi-dimensional configuration space. The particle set, initially widely and evenly spread, becomes drawn first globally towards the global function maximum, and then explores that area of configuration space in increasing detail, able to jump out of

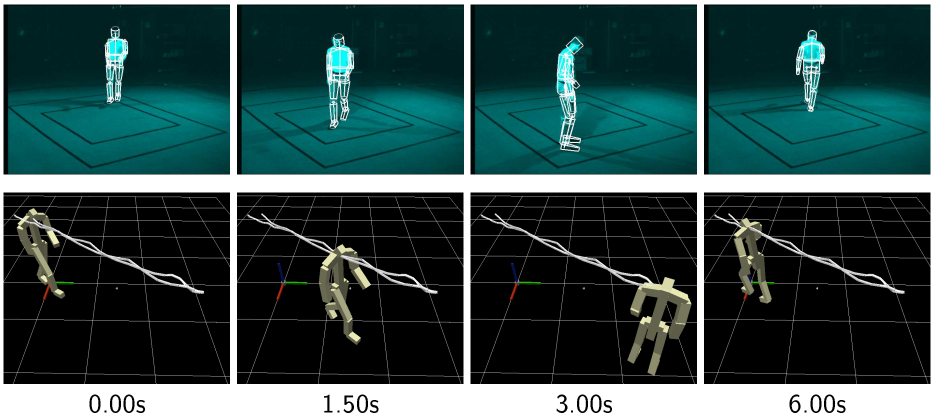


Fig. 5. Walking and turning sequence: the subject walks in a straight line, turns, and walks back. In the upper row, the tracked model template is shown superimposed on the raw images from one of the three cameras used. The lower row shows synchronised animation of a character seen from a viewpoint different to that of any of the input cameras; the waving line seen is the trajectory of the character's root segment, defined at the neck.

local maxima. The parameters of the process (in particular the number of layers M, the number of particles N and the annealing rate, which determines the rate of convergence) must be adjusted such that accurate tracking can be performed with a minimum amount of computational cost; in typical operation, 10–30 layers were used with 200–500 particles. The annealing rate is closely related to the particle survival rate α of MacCormick and Blake [12], affecting the number of particles which are usefully propagated from one layer to the next.

5 Results

We present results from three image captures of a subject wearing normal clothing moving freely within an workspace area of around 5×3 metres. Images were taken from three cameras spaced horizontally at regular intervals around this area, and capture was at 60Hz. Calibration to determine the positions of cameras and their internal parameters to a high accuracy was obtained via a commercial marker-based system.

Image processing was carried out offline using both an SGI Octane workstation and a 1GHz Pentium PC (the two having similar performance), and required around 30 seconds for each frame of video (a rate corresponding to 30 minutes per second of video footage). Although this processing is still a large factor away from the long-term target of real-time, it compares very favourably with other systems in the literature. The sequences tracked in this paper are only a few seconds long, but this shortness is due more to difficulties with managing the very large amounts of image data involved (10 seconds of uncompressed footage from 3 cameras fills a CD-R disc), and the processing time required, than to limitations of the tracker. See the web site given in the abstract of this paper for movie files of these results.

In the first image sequence (Figure 5), the subject walks in a straight line, turns through 180°, and walks back in a total movement of just over 6 seconds. The most challenging part of this sequence from a tracking point of view is the sharp turn, when the subject's arms and legs are close together and likely to be confused, and some tracking inaccuracies are observed here. The second sequence (Figure 6) shows the subject running and dodging in a triangular pattern over nearly 3 seconds. Although tracking rapid movement of this type is generally more difficult due to the larger frame-to-frame displacements involved, the subject's limbs are generally away from his body in this example and could be localised accurately. The third sequence (Figure 7) emphasizes the lack of movement constraints imposed in our tracking framework by tracking a handstand motion. The character model used in this last sequence was augmented with extra degrees of freedom in the back to capture the bending involved.

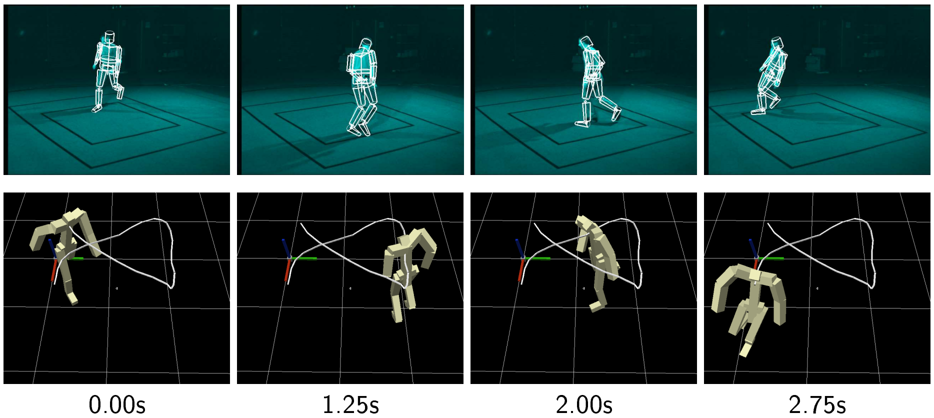


Fig. 6. Running and dodging sequence: the subject runs in a roughly triangular pattern, changing direction abruptly.

6 Conclusions

We have presented convincing results from a system able to perform robust visual motion capture in 3D without artificial markers, and shown that the data obtained can be used to animate a CG character. Annealed particle filtering is able to cut efficiently through the high-dimensional configuration space of an articulated model to recover varied and agile motions without the need for restrictive dynamical models.

Acknowledgements: This work was supported by Oxford Metrics and EPSRC grant GR/M15262.

References

1. A. Baumberg and D. Hogg. Generating spatiotemporal models from examples. In *Proc. British Machine Vision Conf.*, volume 2, pages 413–422, 1995.
2. C. Bregler and J. Malik. Tracking people with twists and exponential maps. In *Proc. CVPR*, 1998.
3. J. Deutscher, A. Blake, B. North, and B. Bascle. Tracking through singularities and discontinuities by random sampling. In *Proc. 7th Int. Conf. on Computer Vision*, volume 2, pages 1144–1149, 1999.

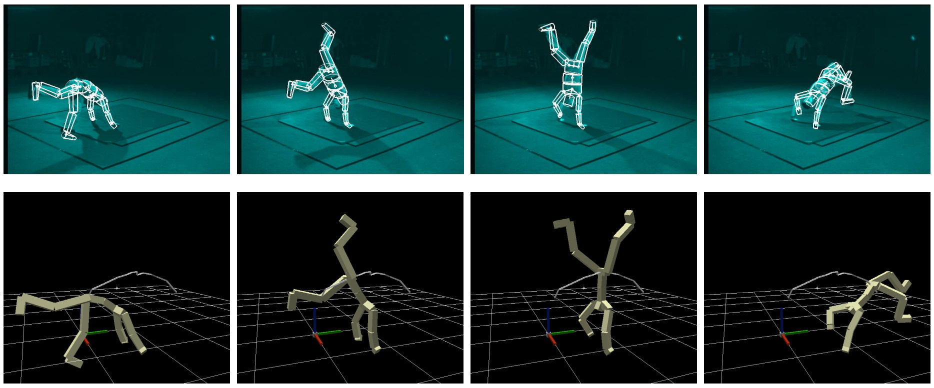


Fig. 7. Handstand sequence: the subject executes a poorly-balanced handstand, toppling rapidly over. In this sequence the trajectory shown is that of the base of the spine.

4. J. Deutscher, A. Blake, and I Reid. Articulated body motion capture by annealed particle filtering. In *Proc. Conf. Computer Vision and Pattern Recognition*, volume 2, pages 1144–1149, 2000.
5. D. Gavrila and L.S. Davis. 3d model-based tracking of humans in action: a multi-view approach. *Proc. Conf. Computer Vision and Pattern Recognition*, pages 73–80, 1996.
6. I. Haritaoglu, D. Harwood, and L. Davis. w^4s: A real-time system for detecting and tracking people in 2.5D. In *Proc. 5th European Conf. Computer Vision*, volume 1, pages 877–892, Freiburg, Germany, June 1998. Springer Verlag.
7. C. G. Harris. Tracking with rigid models. In A. Blake and A. Yuille, editors, *Active Vision*. MIT Press, Cambridge, MA, 1992.
8. Nicholas R. Howe, Michael E. Leventon, and William T. Freeman. Bayesian reconstruction of 3D human motion from single-camera video. In *Advances in Neural Information Processing Systems 12*, pages 820–826. MIT Press, 2000.
9. M.A. Isard and A. Blake. Visual tracking by stochastic propagation of conditional density. In *Proc. 4th European Conf. Computer Vision*, pages 343–356, Cambridge, England, Apr 1996.
10. S.X. Ju, M.J. Black, and Y. Yacoob. Cardboard people: A parameterized model of articulated motion. In *2nd Int. Conf. on Automatic Face and Gesture Recognition, Killington, Vermont*, pages 38–44, 1996.
11. D.G. Lowe. Robust model-based motion tracking through the integration of search and estimation. *Int. J. Computer Vision*, 8(2):113–122, 1992.
12. J. MacCormick and A. Blake. Partitioned sampling, articulated objects and interface-quality hand tracking. In *Accepted to ECCV 2000*, 2000.
13. D. Reynard, A.P. Wildenberg, A. Blake, and J. Marchant. Learning dynamics of complex motions from image sequences. In *Proc. 4th European Conf. Computer Vision*, pages 357–368, Cambridge, England, Apr 1996.
14. H. Sidenbladh, M. J. Black, and D. J. Fleet. Stochastic tracking of 3D human figures using 2d image motion. In *Proceedings of the 6th European Conference on Computer Vision, Dublin*, 2000.
15. R. A. Smith, A. W. Fitzgibbon, and A. Zisserman. Improving augmented reality using image and scene constraints. In *Proc. 10th British Machine Vision Conference, Nottingham*, pages 295–304. BMVA Press, 1999.
16. Vicon web based literature. URL http://www.metrics.co.uk, 2001.