Modelling and Tracking Articulated Motion from Multiple Camera Views

Maurice Ringer and Joan Lasenby
Cambridge University, Department of Engineering
Trumpington Street, Cambridge CB2 1PZ, UK
mar39/jl@eng.cam.ac.uk
www-sigproc.cam.ac.uk/vision

Abstract

This paper describes a scheme for modelling and tracking the motion of articulated bodies using a number of video cameras. The aim is to obtain complete and accurate information on the three-dimensional location and motion of the bodies over time. Applications include medicine, sports analysis and motion capture for animation.

Feature extraction is avoided by placing markers at the joints of the body so that model selection, marker-to-measured-point association, occlusion and the choice of tracking filter are the important issues. While the scheme is general for any number of cameras, emphasis is placed on systems with a small number of cameras where occlusions are a major problem. The system is an amalgamation of new ideas and existing techniques drawn from a variety of disciplines such as machine vision, geometric algebra and radar tracking theory, which have been extended and developed for the marked joints/multiple camera problem. The proposed schemes for modelling and tracking would be easily adapted to markerless motion capture.

The paper concludes with examples of the system successfully tracking limb motion using three cameras.

1 Introduction

The problem we address here is to determine and track parameters of an articulated body moving through a sequence of video images. This *motion capture* is most commonly applied to human subjects and is used for a variety of purposes, including medical investigation, sports analysis and animation. For example, the performance of a runner could be quantified by knowing how the location and velocity of his legs vary during a filmed race. Likewise, similar information can be extracted from a video of a person walking in order to detect gait irregularities and to investigate possible causes.

The proposed system uses markers placed at the joints of the arm(s) or leg(s) being analysed, although much of the scheme would be equally applicable to markerless motion capture. The location of these markers on each camera's image plane provide the input to the tracking system with the result that the required parameters of the body can be estimated to far greater accuracy than one could obtain in the markerless case. Techniques for markerless measuring are usually based either on edge detection or colour flow fields, the

BMVC 2000 doi:10.5244/C.14.18

performance of which vary greatly depending on lighting conditions, background complexity and occlusions. The accuracy of such techniques is not currently sufficient for biomechanical and medical analyses.

The system we will describe uses techniques for tracking and 3D reconstruction of location which use all of the available data in all of the cameras. As such we can avoid many problems which may occur when markers are occluded or joints become close or cross over. In such cases, we wish to avoid both time consuming user interaction to guide the tracker, or the introduction of more cameras. The aim is to provide a mathematically optimal solution to marked-limb motion capture resulting in an automated low cost system with high accuracy. Currently, the proposed system is implemented as an off-line processor, however its execution time is comparable to the video sequence length and with the correct computer it is likely that the system would be capable of tracking human motion in real-time.

The proposed system is presented in three sections. Section 2 outlines the model and its parameters, how they are expected to change over time and how they are related to the measurements in the cameras. Section 3 describes a method of estimating the parameters of this model given a sequence of measurements, and focuses particularly on the problem of associating the markers being tracked with the points detected by the cameras. Section 4 shows the results when the model and techniques are used to track limb-motion in real subjects. The final section then summarises the performance of the system and proposes possible extensions.

2 The Model

Our aim is to estimate and track a number of parameters of an articulated body. Prior information on these parameters is contained in the model, which is composed of two parts: the kinematic model and the measurement model.

2.1 Kinematic Model

The model chosen consists of a series of constrained linked rigid rods so that the location and position of the articulated body is fully described by an origin point in space, a, the lengths of each rod, l_i, and the angles which relate each rod to its neighbours, θ_i.

In this section and the next, it will be assumed that a single leg is being viewed and tracked. We will also show how such a model can be extended to include other limbs, thus building up a model of any desired complexity.

The parameters which fully describe the location and position of a leg are shown in figure 1. In this model, the mid-point of the hips represents the reference point $a = (X_a, Y_a, Z_a)$. The hip is able to rotate freely about a in three dimensions (a movement described by two degrees of freedom, since at this link we are not concerned with rotation about the axis of the rod) and the knee is able to move about the hip with three degrees of freedom (where now we do model the rotation of the upper leg around its own axis). Finally, the ankle and toe are modelled such that the allowed movements keep the links representing the upper leg, lower leg and foot in the same plane.

When tracking parameters whose temporal kinematics are unknown, it is common practice to assume the parameters follows a linear trajectory over small durations of time [1, 2, 13].

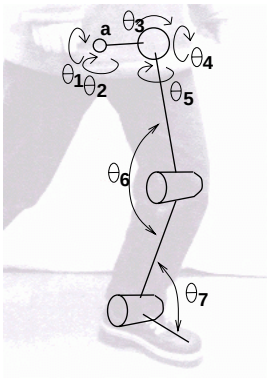

Figure 1: Parameters used to describe the position of a leg

We adopt this model and thus the evolution of the angular parameters is given by:

$$\begin{bmatrix} \theta_i(k+1) \\ \dot{\theta}_i(k+1) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \theta_i(k) \\ \dot{\theta}_i(k) \end{bmatrix} + \begin{bmatrix} 0.5 \\ 1 \end{bmatrix} v_i(k) \tag{1}$$

where the dotted variables represent the rate of change of the given angular parameter, v_i represents zero-mean process noise, and k is the frame number (or an index into discrete time). The variance of v_i determines how much acceleration the angular parameters are expected to undergo from one frame to the next.

This model can also be considered as a simplification of the AR process proposed by [11]. In our case, however, no training is required and the process is capable of describing any motion.

The reference point a is assumed fixed, thus the equation describing its evolution through time is

$$a(k+1) = a(k) + v(k) \tag{2}$$

where v is, once again, process noise. In this case, the variance of v corresponds to how likely it is that the position of a changes from one time frame to the next.

We now define a state, $\mathbf{x}$, which contains all the parameters which define the location and velocity of a leg in time, $(a, \{\theta_i\})$. Note that the state does not include the lengths of the rods, as they are assumed known.

Combining the above equations produces the kinematic equation:

$$\mathbf{x}(k+1) = F\mathbf{x}(k) + G\mathbf{v}(k) \tag{3}$$

where F and G are constant matrices described by equations (1) and (2).

Similar sets of parameters can be used to describe the location and velocity of other limbs.

State vectors containing information on different limbs could be stacked to form a single state vector and the F and G matrices stacked diagonally to form the new coefficients to the kinematic equation.

Many current systems [3, 9] do not attempt to model kinematic motion and instead rely on the process noise, v_i, to account for the change in limb position from one frame to the next.

Other motion capture systems use complex models in which further parameters are required to define the state's expected trajectory through time [11, 3]. These extra parameters are contained in the matrices F and G and often need to be learnt from a series of training data. Such models are therefore restrictive because they limit the types of motion which can be tracked. To overcome this limitation, these systems are usually forced to use mutiple models and include a mechanism for determining which model is most valid at any given time.

We believe that the model proposed here is a good compromise between the two existing modelling techniques.

2.2 Measurement Model

Each new video frame provides the system with a list of 2D points representing the detections of the markers on the image planes of the cameras. These points are related to the state via the measurement equation:

$$\mathbf{y}(k) = H_k(\mathbf{x}(k)) + \mathbf{w}(k) \tag{4}$$

where $\mathbf{w}$ is a zero-mean random variable representing noise at the detection process and $\mathbf{y}$ is a stacked vector of detected points.

The function H is dependent on k because not every marker may be detected in each frame and camera. As the number of detected markers vary, so does the length of $\mathbf{y}$ and thus the dimensions of H. Determining which detections correspond to the markers being tracked, and therefore the contents of $\mathbf{y}$, is the association problem and is discussed in detail in the following section.

The contents of H was generated using geometric algebra (GA) [7] and then converted to the Euler-angle formulation using the computer-algebra package Maple. If the model expands (for example, if further limbs are added) then GA provides a framework in which H can be derived simply. Also, in order to implement the extended Kalman filter (discussed in the following section), the Jacobian of H is required — a complex task should H be written in terms of vectors and Euler angles, but one which is simple using GA.

3 Tracking

The objective of the tracking filter is to estimate the state at any given time, $\mathbf{x}(k)$, given the measurement at that time and all measurements prior to it. Two methods are discussed in this section: the extended Kalman filter (EKF) [1] and the currently popular concept of the Bayesian particle filter [6, 8]. Most applications of these filters to date have been to single camera data. Before either method can be employed, however, it is necessary to solve the *association problem*.

3.1 Association

At a given time, k, each of the M cameras viewing the articuled body detects Q_m^k ($m = 1 \ldots M$) bright points. These points are due to either a marker placed on the body or an error in the detection process, for example, if a background light or reflection is interpreted

as a marker. Also, a particular marker may not have been detected by every camera, either because it was occluded from the camera's view or because the detection process failed.

Thus there exists a requirement to determine which detected points in each camera correspond to each of the N markers being tracked. This forms the association or correspondence problem and its solution as presented here is a combination of techniques used by radar engineers [10, 2] and single camera motion capture systems [4], extended to the multiple-camera motion capture problem. We believe this technique for multiple point association in multiple cameras to be novel.

Let Ψ_i^k be a possible combination of correspodences of markers-to-measured-points at time k. Index i runs over the set $\{1 \ldots I_k\}$, where I_k is the total number of possible combinations:

$$I_k = \prod_{m=1}^{M} \left(\sum_{p=0}^{\min(N,Q_m^k)} {}_N P_p \; {}_{Q_m^k} C_p \right) = \prod_{m=1}^{M} \left(\sum_{p=0}^{\min(N,Q_m^k)} \frac{N! Q_m^k !}{p!(N-p)!(Q_m^k - p)!} \right) \quad (5)$$

where ${}_nP_k$ denotes the permutations of k from n and ${}_nC_k$ denotes the combinations of k from n. The set Ψ_i^k is composed of three elements:

- ϕ_i^k, the list of markers detected in each camera and the corresponding detection. Let $\phi_i^k(j,m)$ be the jth {maker,detected point} pair from camera m given the combination ψ_i ($j = 1 \ldots J_m^k$, where J_m^k is the number of markers that were detected in camera m at time k).
- ζ_i^k, the list of markers that were not detected. Let $\zeta_i^k(p,m)$ be the pth marker missing from camera m ($p = 1 \ldots (N - J_m^k)$) at time k.
- ξ_i^k, the list of detected points that do not correspond to any markers being tracked. Let $\xi_i^k(q,m)$ be the qth detected point in camera m at time k which belongs to this set ($q = 1 \ldots (Q_m^k - J_m^k)$).

By extending the argument proposed in [10], it can be shown that

$$\begin{aligned} P(\Psi_i^k) \quad = \quad & \frac{1}{c} P(\Psi_i^{k-1}) \left(\prod_{n=1}^{N} P_R(\phi_i^k(n,\cdot)) \right) \prod_{m=1}^{M} \left(\prod_{j=1}^{J_m^k} P_C(\phi_i^k(j,m)) \right. \\ & \left. \prod_{p=1}^{N-J_m^k} \left(1 - P_D(\zeta_i^k(p,m))\right) \prod_{q=1}^{Q_m^k - J_m^k} P_{FA}(\xi_i^k(q,m)) \right) \end{aligned} \quad (6)$$

where c is a normalising constant, $P_C(j)$ is the probability that the association of marker to detected point is correct, $P_D(p)$ is the probability that marker p was detected, $P_{FA}(q)$ is the probability that point q was a false detection, and $P_R(j)$ is the probability that the list of points j detected in each camera originated from the same point in space.

We desire a method of determining $\hat{\Psi}_i^k$, which maximises this expression, given that it is computationally too expensive to evaluate $P(\Psi_i^k)$ for every i. For example, if each of three cameras tracking a pair of legs detected seven markers ($N = 8$, $M = 3$ and $Q_m = 7$ for all m), the total number of combinations is 6.13×10^{16}!

Instead, the B most likely combinations of marker-to-detected-point correspondences for each camera, $P(\Psi_b^k(m))$, are determined separately, resulting in B^M possible combinations whose probabilities are given by

$$P(\Psi_{i'}^k) = \frac{1}{c}\left(\prod_{j=1}^{J_m^k} P_R(\phi_{i'}^k(j,\cdot))\right)\left(\prod_{m=1}^{M} P(\Psi_{b_m}^k(m))\right) \tag{7}$$

where i' is an index into the B^M possible combinations and b_m is the particular combination of marker-to-detected-point correspondences for camera m ($b_m = 1 \ldots B$) given by i'. The B most likely combinations of marker-to-detected-point correspondences for each camera is determined by maximising

$$\begin{aligned} P(\Psi_b^k(m)) \quad = \quad & \left(\prod_{j=1}^{J_m^k} P_C(\phi_b^k(j,m))\right)\left(\prod_{p=1}^{N-J_m^k}\left(1 - P_D(\zeta_b^k(p,m))\right)\right) \\ & \left(\prod_{q=1}^{Q_m^k - J_m^k} P_{FA}(\xi_b^k(q,m))\right) \end{aligned} \tag{8}$$

which presents a considerably smaller search space than does equation (6).

The probability that a given marker-to-detected-point correspondence is correct, $P_C(\phi_b^k(j,m))$, is the likelihood that the distance, d_1, between the detected point and the expected position of the marker on the camera's image plane is zero. In the implementation described in section 4, d_1 was taken to be Gaussian distributed with variance equal to that of $\mathbf{w}(k)$, the measurement noise.

The probability that a point is a false detection, P_{FA}, is assumed constant. In many applications of these techniques, P_D, the probability that a marker is detected, is also usually assumed constant. Here, however, we use the estimates of the marker locations in space, the knowledge of how the markers are linked and the camera positions and orientations to determine the likelihood that a given marker is occluded in a particular camera. A marker is determined to be possibly occluded by a linked rod according to whether it falls behind (when viewed from the camera origin) the plane formed by the rod and the shortest line between the rod and the line of projection of the marker to the camera plane. The probability of marker detection is then

$$P_D(p) = \begin{cases} P_D' & \text{if } p \text{ is not occluded} \\ P_D' - \exp(-d_2^2/\sigma^2)/\sqrt{2\pi\sigma^2} & \text{if } p \text{ is occluded} \end{cases} \tag{9}$$

where d_2 is the distance between the occluding rod and the line of projection of the marker to camera plane, σ^2 is a measure of the width of each rod and P_D' is the probability that a marker not occluded by any limb is detected by the feature extraction procedure. It is assumed that P_D' is constant. An example of the calculation of P_D appears in figure 2. The computational requirements of this technique are improved further by only considering those combinations in each camera in which the detected point falls within a fixed region around the expected location of the marker on that camera's image plane. That is, those $\{$marker, detected point$\}$ pairs for which d_1 is less than a constant threshold. This approach is analogous to the clustering technique used by radar engineers [10, 2].

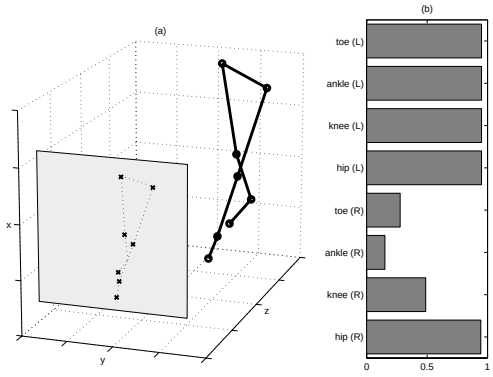

Figure 2: (a) Example estimate of the location of a pair of legs, the projection of the markers onto one camera's image plane (the dotted line) and the points detected by that camera (the $\times$'s); and (b) the probability of detection, P_D, for each of these markers ($P'_D = 0.95$)

3.2 Extended Kalman filter

Determining the most likely combination of marker-to-detected-point correspondences gives the construction of the measurement vector, $\mathbf{y}(k)$. An estimate of the state, $\hat{\mathbf{x}}(k)$, can then be calculated using the extended Kalman filter (EKF) update equations [1]:

$$\hat{\mathbf{x}}(k) = \hat{\mathbf{x}}(k-1) + W(k)(\mathbf{y}(k) - F\hat{\mathbf{x}}(k-1)) \tag{10}$$

where

$$W(k) = P'(k)\, h_k^T \left(h_k P'(k) h_k^T + R\right)^{-1} \tag{11}$$

$$P'(k) = FP(k-1)F^T + Q \tag{12}$$

$$P(k) = P'(k) - W(k)\left(h_k P'(k) h_k^T + R\right) W(k)^T \tag{13}$$

h_k is the jacobian of H_k evaluated at $Fx(k-1)$, Q is the covariance of the process noise, $\mathbf{v}(k)$, and R is the covariance of the measurement noise, $\mathbf{w}(k)$.

3.3 Particle filtering

Use of the EKF implies our state estimate has a Gaussian distribution (of mean $\hat{\mathbf{x}}(k)$ and covariance $P(k)$). As the measurement function, H_k, is non-linear, it is very unlikely that the estimate of the state follows this distribution, even if the noises in the system ($\mathbf{w}$ and $\mathbf{v}$) do. By way of contrast, *particle filters* attempt to fully describe the distributions of variables of non-linear processes and as such have been used widely in radar tracking [6, 13], single camera computer vision [8] and other automated control (for example, [5]).

Particle filters work by propogating a number of points in state space (the particles) through equations (1) and (2), weighting the samples using the measurement and re-sampling the resulting distribution. In this manner, the distribution of the samples represent the distribution of the estimate of the state, the mean or mode of which can be used as the best estimate of the parameters of the articulated body at any given time.

Weighting the samples by the measurement involves calculating the likelihood function, $P(\mathbf{y}|\mathbf{x})$. In the system we describe here this function is a multivariate Gaussian distribution in measurement space with mean $\mathbf{y}(k)$ (constructed from Ψ_i^k) and covariance R. An obvious extension of this technique is to use a likelihood function with multiple modes to cater for the situation when the most likely marker-to-detected-point correspondence, Ψ_i^k, was not the correct one. To create a likelihood function of n modes, the n most likely correspondence combinations would be determined from equation (6). The measurement vectors constructed from these combinations would form the location of the modes and the probability of the combination would form their relative magnitudes. This extension has been suggested to solve the problem of tracking multiple aircraft by radar [12].

Particle filters are far more computationally demanding than EKFs, although have the advantage of being able to incorporate further constraints in the model by limiting elements of the state space to particular values. For example, the knee cannot hyper-extend and samples which violate this can be suppressed.

4 Results

The system described was implemented and tested on a number of subjects. In each case, three cameras were used and the system successfully chose the correct marker-to-detected-point correspondences and produced a track of the location of the limbs through time.

Figure 3 shows the output of the tracking system when it was used to follow both legs of a person walking. Figure 4 shows similar output when the person is performing a tap dance. In both cases, one leg passes in front of the other and many marker occlusions occur. The tracking system, however, was still able to accurately determine the location of the markers in 3D space. Both of these figures show the output when the extended Kalman filter was used to perform the state estimate updates.

Figure 5 shows the output of the tracking system when it is used to follow the arms of a golfer taking a swing. This figure shows the resulting track when the particle filter was used (with 500 samples). The histograms of the samples at an example time for this analysis are also shown in this figure. It can be seen from these histograms that the estimate of the state is approximately single mode and its distribution appears close to Gaussian. This confirms the validity of the EKF and explains why it was able to successfully track all test cases.

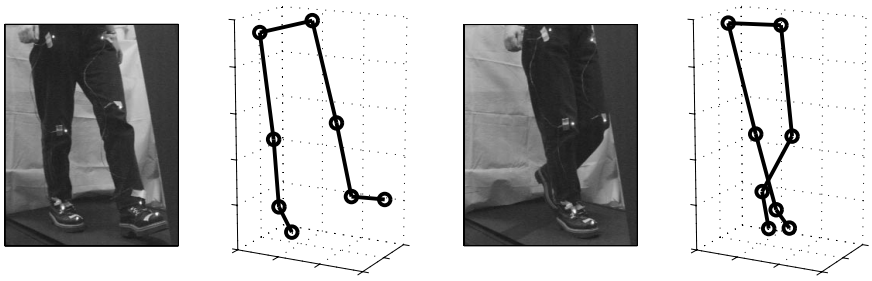

Figure 3: Example of the proposed system tracking the legs of a person walking

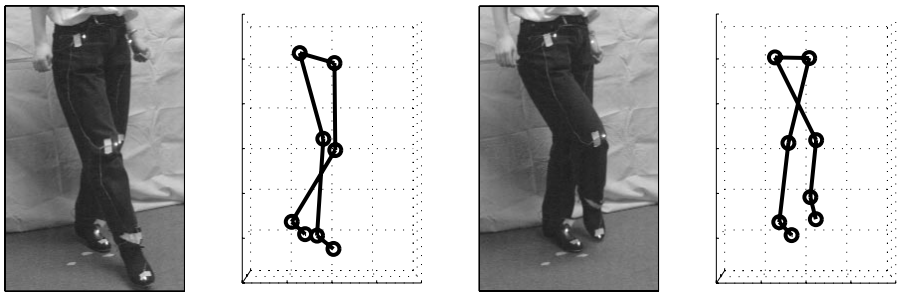

Figure 4: Example of the proposed system tracking the legs of a tap dancer

The particle filter took approximately four times as long to process each video sequence compared to the EKF, whose execution time was almost real-time. Also, the estimates of the locations of the joints as produced by the particle filter contained more noise than those produced by the EKF (in the case of the particle filter, the mean of the samples was used as the state estimate).

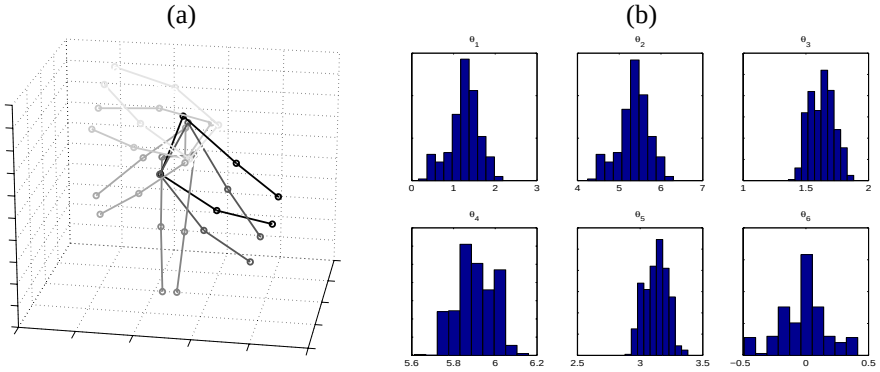

Figure 5: (a) Example of the proposed system tracking the arms of a golfer taking a swing (the darker colours represents states at later times); and (b) Histograms of the elements of the samples of the particle filter which parameterise the left arm.

5 Conclusions

A new scheme for modelling and tracking the location and motion of articulated bodies using multiple cameras has been presented. In particular, we use novel methods for modelling the kinematic motion and a new technique for calculating the marker-to-detected-point correspondences.

The extended Kalman filter and particle filter were used to update the state estimates.

Both techniques proved successful, being able to track all test limbs applied to them thus far; however the EKF was preferred because its computational demands were less and there was no evidence of the state estimates being significantly non-Gaussian.

The system is currently general and works for any number of points and cameras and any model. A future extension will be to extend the tracking algorithms to deal with multiple EKFs or multi-modal particle filters, as this would be necessary for very complex motions.

References

[1] Y. Bar-Shalom, editor. *Multitarget-multisensor tracking*. Artech House, 1992.

[2] S. Blackman and R. Popoli. *Design and analysis of modern tracking systems*. Artech House, 1999.

[3] A. Blake and M. Isard. *Active Contours*. Springer-Verlag, 1998.

[4] J. Dezert. Tracking maneuvering and bending extended targets in cluttered environment. In *Proc. SPIE conf on Signal and Data Proc of Small Targets*, pages 283–293, USA, Apr 1998.

[5] D. Fox, W. Burgard, F. Dellaert, and S. Thrun. Monte carlo localization: Efficient position estimation for mobile robots. In *Proc. of AAAI-99*, 1999.

[6] N. Gordon, D. Salmond, and A. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. In *IEEE Proc F*, number 140, pages 107–113, 1993.

[7] D. Hestenes and G. Sobczyk. *Clifford Algebra to Geometric Calculus: A unified language for mathematics and physics*. Dordrecht, 1984.

[8] M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *Proc. European Conf. on Computer Vision*, volume 1, pages 343–356, 1996.

[9] J. Rehg and T. Kanade. *Proc. of Third European Conf. on Computer Vision*, volume 2, chapter Visual Tracking of High DOF articulated structures: an application to human hand tracking, pages 35–46. Springer-Verlag, 1994.

[10] D. Reid. An algorithm for tracking multiple targets. In *IEEE Trans on Automatic Control*, volume AC-24, pages 843–854, Dec 1979.

[11] J. Rittscher and A. Blake. Classification of human body motion. In *Proc. Int. Conf. Computer Vision*, 1999.

[12] D. Salmond and N. Gordon. Group and extended object tracking. In *Target Tracking: Algorithms and Applications*, London, UK, Nov 1999.

[13] L. Stone, C. Barlow, and T. Corwin. *Bayesian multiple target tracking*. Artech House, 1999.