Abstract—For improving the efficiency of human 3D tracking, we present an algorithm to track 3D Arm Motion. First, the Hierarchy Limb Model (HLM) is proposed based on the human 3D skeleton model. Second, via graph decomposition, the arm motion state space, modeled by HLM, can be decomposed into two low dimension subspaces: root nodes and leaf nodes. Finally, Rao-Blackwellised Particle Filter is used to estimate the 3D arm motion. The result of experiment shows that our algorithm can advance the computation efficiency.

Keywords—Hierarchy Limb Model; Rao-Blackwellised Particle Filter; 3D tracking

I. INTRODUCTION

ALTHOUGH caught the many researchers’ attention because of its widely application, human 3D tracking is still a challenging task because of the exponentially increased computational complexity in terms of the freedom degrees.


In recent years, Rao-Blackwellised Particle Filter [10] [11] [12] has been widely studied in human tracking. Xinyu [12] et al. learns motion correlation using the Partial Least Square, and proposed the RBPF-PLS algorithm to track the walking pose. Although these algorithms have achieved the goal, they can’t track any motion in the nature scene but tracking the learned motion. And learning a general probabilistic model in full space is very difficult because of the high dimensionality and the huge amounts of training data to account for motion complexity. Because of the restrictive dependency, RBPF can not be directly used for 3D human tracking.

Focused on these problems, the paper proposes the 3D arm motion tracking algorithm using Rao-Blackwellised Particle Filter based on Hierarchy Limb Model. The arm motion spaces can be divided into two parts: root variables and leaf variables by the Hierarchy Limb Model. Rao-Blackwellised Particle Filter can be used to track arm motion via the decomposition. As a result, our algorithm can advance the computational efficiency because of the lower dimensionality of the search space and the reduced amounts of particle.

The paper is organized as follows. Section 2 describes the Hierarchy Limb Model. The 3D arm tracking algorithm based on RBPF is proposed in Sention 3. Experimental results and analysis are shown in Section 4, and finally concludes the paper.

II. HIERARCHY LIMB MODEL

The arm can be represented by an arm graphical model such as shown Fig. 1(a). The circle nodes corresponds to a part of right arm, such as the right upper arm and the right lower arm. The square nodes are the observation values associated with each circle nodes. The directed links represent physical constraints among different parts of the right arm. The directed link from a part’s state to its associated observation represents the local observation likelihood. In order to describe the motion of an articulated object, we accommodate the state dynamics by a dynamical graphical model such as shown in Fig. 1(b). It contains two consecutive time frames. The directed links between consecutive states represents the dynamics translation from time \(t-1\) to time \(t\).

We model the arm as 2 cylinders connected at revolute joints, and denote the state of each part of arm at time \(t\) by \(x_i\), where \(i=1,2\) is the index of two parts of arm. The arm motion state space is 12-dimensional, including 6D for the global (shoulder) position and orientation, 3D for the elbow as \(x_{i_2}\), 3D for the wrist as \(x_{i_3}\).
According to the characteristics of arm motion, the motion of any node of the arm only interacts with its children nodes. For example, the motion of lower arm is not constrained by any limbs but only the motion of corresponding upper arm. Fig. 2 (a) show the decomposition result for the right arm in Fig. 1 (b), and Fig. 2 (b) is the associated moral graph via the separation theorem and the characteristics of the dynamic Markov network [13].

Fig. 2 Arm graphical model decomposition. (a) The decomposition of dynamical graphical model; (b) Corresponding moral graph.

Using the arm hierarchy model, the problem of tracking right arm motion can be formulated as the prediction of \( x_t \) at time \( t \).

### III. 3D ARM TRACKING WITH RBPF

By the HLM, arm motion can be decomposed into two parts: \( x_u \) (upper arm) and \( x_l \) (lower arm). We denote \( x_u \) as root variable \( R_u \), and \( x_l \) as leaf variable \( L_l \). In the algorithm, we marginalize the lower arm pose variable by using the motion speed correlation matrix \( V \). The strategy of Rao-Blackwellised Particle Filter is to partition the full state space into two parts, \( R_u \) (root variable) and \( L_l \) (leaf variable), so the \( P(L_l | R_u, Z_{\text{vu}}) \) is a distribution that can be computed exactly conditional on \( R_u \), and the distribution \( P(X_u | Z_{\text{vu}}) \) can be estimated using sample methods, such as Markov Chain Monte Carlo.

The posterior probability distribution for the right arm motion can be factorized by Equation (1).

\[
P(R_u, L_l | Z_{\text{vu}}) = P(L_l | R_u, Z_{\text{vu}})P(R_u | Z_{\text{vu}}) \tag{1}
\]

Given the speed correlation matrix \( V \), and the image measurements \( Z_{\text{vu}} \), the Equation (1) can be approximated by the following expression:

\[
P(R_u, L_l | Z_{\text{vu}}) = P(L_l | R_u, Z_{\text{vu}})P(R_u | Z_{\text{vu}})
\]

In Equation (2), the integral parts is propagated the root variables \( R_u \) utilizing the state transition model according to the standard Particle Filter. The leaf variables \( L_l \) can be predicted analytically using Kalman Filtering prediction. The image likelihood model, as \( P(Z_{\text{vu}} | R_u, L_l) \), is computed in the state space, including root and leaf variable.

#### A. State Transition Model

In particle filter theoretical framework, the state transition model, by which particle is generated, is described as shown Equation 3.

\[
x_i = x_{i-1} + v_i, \; v_i \sim N(\mu, \Sigma) \tag{3}
\]

Where \( v_i \) is drawn from the Gaussian noise that the expectation \( \mu \) is a \( 3 \times 1 \) scalar, which is defined as the motion speed of root variables, and the variance \( \Sigma \) is the \( 3 \times 3 \) diagonal matrix. Utilizing hard prior, we can eliminates the particles corresponding to implausible arm pose to reduce the search space.

#### B. Prediction for the Leaf Variables

The leaf variables can be predicted by the use of the motion speed correlation matrix \( V \) using the Kalman Filter while the root variables are propagated. Condition on the root variables and the correlation matrix \( V \), the leaf variable \( L_l \) are described as the following expression:

\[
L_{l} = AL_{l} + V(R_{\text{u}} - \hat{R}_{\text{u}}) + \xi_{l}, \; p(\xi) \sim N(0, \Phi) \tag{4}
\]

\[Z_{\text{vl}} = H L_{l} + \zeta_{l}, \; p(\zeta) \sim N(0, \Psi)\]

Where \( \xi \) represent the process noise and \( \zeta \) represent the measurement noise. \( V \) is defined as the motion speed correlation matrix. Then the leaf variables are predicted by Equation (5).

\[
L_{l} = AL_{l} + V(R_{\text{u}} - \hat{R}_{\text{u}}) + \xi_{l}, \; p(\xi) \sim N(0, \Phi) \tag{5}
\]

\[P_{l} = AP_{l-1}A^T + \Phi\]

In Equation (5), \( i \) is defined as the particle index. In Equation (4) and Equation (5), \( A \) and \( H \) are assumed to be the diagonal matrix.

According to the Equation (4) and Equation (5), Kalman Updates can be computed as the following expression:
\[ K'_t = P'_t H' (H P'_t H'^T + \Psi) \]
\[ L'_t = L'_t + K'_t (z_t - H L'_{t-1}) \]
\[ P'_t = (1 - K'_t H) P'_t \]  

(6)

C. Image Likelihood

The observation likelihood model is represented for the matching relationship between the human appearance model and the features subtracted from the image among the particle filter theoretical framework. In this section, color distribution and image edge information are used to calculate the matching similarity between the human appearance model and the features subtracted from the image.

Color distributions are used as target models as they achieve robustness against non-rigidity, rotation and partial occlusion. The weighted color histogram, which consists of \( m=8 \times 8 \times 8 = 512 \) bins, is chosen and calculated in HSV color space to decrease the effect of the illumination.

The projection quadrilateral of the limb shape is defined as \( Dr \), \( y \) is the point which is the projection of the origin of the local coordinate system on image plane, and color distribution is defined as \( p'_t = \{ p''_{t,j} \} \). For any pixel point \( \tilde{x}' \in Dr \), \( p''_{t,j} \) can be calculated as following expression:

\[ p''_{t,j} = C \sum_{j=1}^{K} \delta \left( \frac{y - \tilde{x}'}{S_{vo}} \right) \delta \left( h(\tilde{x}') - u \right) \]  

(7)

Where \( \delta(x) \) is the delta function, \( S_{vo} \) is the area of \( Dr \), \( C \) is the normalized constant. The Bhattacharyya distance is used to calculate the similarity between two weighted color histograms.

IV. EXPERIMENTAL RESULT AND ANALYSIS

We have done experiments to track the right arm motion using the HumanEva data sets [14]. The experiment chooses the right arm motion color video made in the front to reduce the self-occlusions.

A. Experimental Result

Table 1 is the comparison of mean error, Mean, and error variance, Std. between the ground truth and the prediction value of the right lower arm under different count of particle using our algorithm in X direction, Y direction and Z direction. The Equation (8) is represented for mean error. The Equation (9) is represented for error variance.

\[ \text{Mean} = \frac{\sum_{i=1}^{N} (x'_i - x'_t)}{T} \]  

(8)

\[ \text{Std} = \sqrt{\frac{\sum_{i=1}^{N} (x'_i - \text{Mean})^2}{T}} \]  

(9)

In Equation (8) and Equation (9) the frames of test video is described as \( T \), and \( T=796 \). \( x'_i \) is the prediction value, \( x'_t \) is the ground truth at frame \( t \).

From Table I, the mean error and error variance between the prediction and ground truth have not evidently changes as the particle count of limbs increasing. Then we can draw the conclusion that the count of particle for limbs can not affect the tracking result of our algorithm. Figure. 3 shows the tracking results of 3D arm motion by our algorithm. It is no evidently different between the tracking results of our algorithm and the real pose of arm motion.

B. Experimental Analysis

The count of joints, which need be tracked in each tracking process, is defined as \( K \). Each joint needs \( N_i \) particles to track the joint, \( i=1 \ldots K \). HLMRBPF needs \( \prod N_i \) particles for all limbs and its computational complexity is \( E (N^K) \). While standard particle filter generates \( N^K \) kinds of combination patterns of particle in whole state space, which is formulated as \( N^K \) kinds of motion states and the computational complexity of the standard particle filter is \( E (N^K) \). In our experiment, \( K = 2 \), \( N_i = 200 \), \( N_i = N_i//10 \), \( N = 200 \). Then time-cost of HLMRBPF tracking one every frame image is 5908ms, while the standard Particle Filter (SPF) needs 14768ms.

Table II show the comparison of mean error, Mean, error variance, and Std. between the prediction values using two algorithms and the ground truth in X direction, Y direction, and Z direction.

V. CONCLUSIONS

The paper proposes 3D arm motion fast tracking algorithm. Based on the HLM, the algorithm can transfer the global optimal search of the whole state space to the top-bottom search based on the joints under the case that the dimension of state space is unchangeable. In the process of tracking, the particle count is reduced by the prediction of each joint of HLMRBPF. The experiment shows that the tracking result using our algorithm is not evident difference compared with the standard
particle filter under the same dimension of state space. The algorithm can effectively apply to track 3D arm motion based on Particle Filter.

REFERENCES


Fig. 3 3D animation Comparison between the tracking result by our algorithm and ground truth; (a) 3D animation for the tracking value of our algorithm; (b) 3D animation for the ground truth.