Distributed Human 3D Pose Estimation and Action Recognition*

Guoliang Liu, Tiantian Liu and Guohui Tian
School of Control Science and Engineering
Shandong University
Jinan, Shandong, China
{liuguoliang}@sdu.edu.cn

Ze Ji
School of Engineering
Cardiff University
Cardiff, CF24 3AA, UK
jiz1@cardiff.ac.uk

Abstract—In this paper, we propose a distributed solution for 3D human pose estimation using a RGBD camera network. The key feature of our method is a dynamic hybrid consensus filter (DHCF) is introduced to fuse the multiple view information of cameras. In contrast to the centralized fusion solution, the DHCF algorithm can be used in a distributed network, which requires no central information fusion center. Therefore, the DHCF based fusion algorithm can benefit from many advantages of distributed network. We also show that the proposed fusion algorithm can handle the occlusion problems effectively, and achieve higher action recognition rate compared to the ones using only single view information.

Index Terms—Distributed information fusion, RGBD camera network, consensus filtering, human pose estimation, action recognition.

I. INTRODUCTION

Due to the widespread of the low-cost RGBD sensors, the real time 3D human pose estimation is available on the commercial products, such as Microsoft Xbox using Kinect for human computer interaction (HCI) games [1], [2]. More potential applications have been studied in many research fields. For instance, Morato et al. developed a N-Kinect system to build an explicit model of the human body, which is used to detect an imminent collision between the robot and the human [3]. Pathirana et al. proposed a Kinect based bio-kinematic measurement system for rehabilitation and physiotherapy applications, i.e., monitoring of home-based prescribed exercise routines which can significantly reduce the need for patients to travel to regional centers [4]. Geiselhart et al. estimated the workers movements and interaction with digital object models using multi-depth camera system [5], such that a low cost camera setup to facilitate interaction with virtual environments for planning experts is possible.

* This work is partially supported by the National Key R&D Program of China (#2018YFB1306504), National Natural Science Foundation of China (#61605213, #91748115), Young Scholars Program of Shandong University (#2018WLJH71), the Fundamental Research Funds of Shandong University, and the Taishan Scholars Program of Shandong Province. (Corresponding author: Guoliang Liu).

Fig. 1: The experimental setup uses four Kinect V2 cameras, which have a loop topology and construct a distributed sensor network.

To accurately estimate the human pose is still a challenge problem due to the occlusion problem [6]. Many researchers propose to use multiple 3D cameras to handle occlusion problem by fusing multiple view information. The key problem is how to fuse. Current works mostly employ a centralized network topology, and use a central computer node to fuse information from all sensor nodes. The fusion algorithm can be a simple weighted summation method using skeleton joint tracking status [7], [8], [9], or additional bone length constrain conditions [10]. In order to use the temporal information and get smooth trajectory of skeleton joints, a
Kalman filter or particle filter can be employed [11], [12], [13].

Compared to the centralized network topology, the distributed network topology is more desirable in many application scenarios due to its scalability to a large number of sensors, ease of installation and high tolerance to node failure [14]. Furthermore, each camera can process information and make decision locally, such that the camera nodes become distributed smart agents, which can achieve consensus by exchanging information with neighbor nodes after a number of iterations [15]. Song et al. propose a distributed 2D camera network to track human trajectories and recognize actions using a Kalman consensus filter (KCF) [16], [15]. The KCF was introduced by Olfati-Saber and has been applied in many fields [17], [18]. Kamal et al. proved that the KCF cannot handle information redundant and naive node problems for the camera network, and proposed an information weighted consensus filter (IWCF) to replace the KCF, which shows improved performance for 2D human tracking using a 2D camera network [14]. Wang et al. employ the IWCF to track 3D human skeleton joints using multiple Kinects [19]. Li et al. further prove that the IWCF based human skeleton fusion algorithm can achieve higher action recognition [20], [21].

In this paper, we propose a dynamic hybrid consensus filter (DHCF) based human pose estimation and action recognition algorithm using a 3D RGB-D camera network, which can handle occlusion problem and achieve higher recognition rate. A. Dynamic Hybrid Consensus Filter for Skeleton Fusion

The DHCF is a distributed fusion algorithm, whose core idea is the iterative information exchanges between the sensor node and its neighbors [22], [23]. After a number of iterations, the sensor nodes in the whole network can achieve consensus state about the target. Compared to the IWCF proposed in [14], DHCF can achieve faster convergence rate and have no requirement about the total number of sensor nodes. The distributed camera network used in this paper construct an undirected graph \( G = (C, E) \) where \( C = \{1, 2, 3, \ldots, N\} \) means vertex set that has \( N \) nodes, and \( E \subset \{(i, j) | i, j \subset C\} \) denotes the edge set. We define the

\[
\begin{align*}
\text{Algorithm 1 DHCF based skeleton fusion} \\
\text{• Initialization: total consensus iteration steps } L, \text{ process noise } Q \text{ and measurement noise } R. \\
\text{• For } k = 1, \ldots, \infty: \\
\text{1) Predicted human pose for the next time step:}
\hat{x}_{i,k} = F_k x_{i,k-1} \\
\hat{Y}_{i,k} = (F_k Y_{i,k-1} F_k^T + Q_k)^{-1} \\
\hat{y}_{i,k} = \hat{Y}_{i,k} \hat{x}_{i,k} \\
\text{2) Computation of consensus proposals:}
\begin{cases}
\text{if } i \in S \text{ then} \\
\quad u_{i,k} = H^T_{i,k} R^{-1}_{i,k} z_{i,k}, \quad U_{i,k} = H^T_{i,k} R^{-1}_{i,k} H_{i,k} \\
\quad b_{i,k} = 1 \\
\text{else}
\quad u_{i,k} = 0, \quad U_{i,k} = 0, \quad b_{i,k} = 0
\end{cases}
\text{3) Perform consensus iteratively:}
\begin{align*}
\text{Initialization: } b^0_{i,k} &= b_{i,k}, \quad (\hat{g}^0_{i,k} = \tilde{y}_{i,k}, \hat{Y}^0_{i,k} = \hat{Y}_{i,k}), \quad (u^0_{i,k} = u_{i,k}, \quad U^0_{i,k} = U_{i,k}) \\
\text{for } l = 1 \text{ to } L \text{ do} \\
&\quad (a) \text{ Send } b^{l-1}_{i,k}, \quad (\hat{g}^{l-1}_{i,k}, \hat{Y}^{l-1}_{i,k}), \quad (u^{l-1}_{i,k}, U^{l-1}_{i,k}) \text{ to all neighbors } j \in N_i \\
&\quad (b) \text{ Receive } b^{l-1}_{j,k}, \quad (\hat{g}^{l-1}_{j,k}, \hat{Y}^{l-1}_{j,k}), \quad (u^{l-1}_{j,k}, U^{l-1}_{j,k}) \text{ from all neighbors } j \in N_i \\
&\quad (c) \text{ Update consensus terms}
\hat{y}^{l}_{i,k} = \epsilon_{i,j,k} \sum_{j \in N_i} \hat{y}^{l-1}_{j,k}, \quad \hat{Y}^{l}_{i,k} = \epsilon_{i,j,k} \sum_{j \in N_i} \hat{Y}^{l-1}_{j,k} \\
\hat{u}^{l}_{i,k} = \epsilon_{i,j,k} \sum_{j \in N_i} \hat{u}^{l-1}_{j,k}, \quad \hat{U}^{l}_{i,k} = \epsilon_{i,j,k} \sum_{j \in N_i} \hat{U}^{l-1}_{j,k} \\
\quad b^{l}_{i,k} = \epsilon_{i,j,k} \sum_{j \in N_i} b^{l-1}_{j,k}
\end{align*}
\text{4) Compute the posterior estimation at } k \text{ time step:}
\begin{align*}
u^L_{i,k} &= \begin{cases} \frac{1}{b^L_{i,k}} & \text{if } b^L_{i,k} \neq 0 \\ 1 & \text{otherwise} \end{cases} \\
y_{i,k} &= \hat{y}^L_{i,k} + u^L_{i,k} x^L_{i,k}, \quad Y_{i,k} = \hat{Y}^L_{i,k} + u^L_{i,k} U^L_{i,k} \\
x_{i,k} &= Y^{-1}_{i,k} y_{i,k}
\end{align*}
\end{align*}
\]
neighbor nodes of \( i_{th} \) node as \( N_i = \{ j \in C | i, j \in E \} \). To model the human motion, we use a linear dynamic model as
\[
x(t + 1) = Fx(t) + w
\]
where \( x(t) = (p_{x}(t), p_{y}(t), p_{z}(t), v_{x}(t), v_{y}(t), v_{z}(t))^{T} \) is the state vector including the position and velocity of the human skeleton joints, \( F \) is the linear state transition matrix and \( w \sim N(0, Q) \) is the Gaussian noise with zero mean and covariance \( Q \). The skeleton measurement from each camera node can be used to update the system state of the filter. The measurement model for each joint using the 3D camera is
\[
z(t) = Hx(t) + v
\]
where \( H \) is the linear observation matrix, \( z \) is the predicted measurement of the joint, which has a Gaussian noise \( v \sim N(0, R) \) with zero mean and covariance \( R \). The DHCF filter can use the dynamic model for prediction and the measurement model for state updating.

The dynamic hybrid consensus filter (DHCF) which handles information fusion problem of multiple RGBD cameras is summarized in Algorithm 1. First, the main parameters of the algorithm are initialized, i.e., number of consensus iterations \( L \), process noise \( Q \) and measurement noise \( R \). Second, the DHCF is conducted in four steps for the time step \( k = 1, \cdots, \infty \). The first step is the prediction using the human skeleton motion model, where the prior mean \( x_{i,k} \), information matrix \( Y_{i,k} \) and information vector \( y_{i,k} \) are calculated as shown in (1), (2) and (3) respectively. The second step is the computation of the information contributions \( u_{i,k} \) and \( U_{i,k} \) according to current skeleton measurements as shown in (4). To weight the information contributions, the DHCF uses a distributed estimation of ratio factor \( S/N \) to solve the naive node and overweighting problems, where \( S \) is the number of cameras that have valid measurements of human target. We introduce a quantity \( b_{i,k} \) which indicates whether the camera is a naive node, i.e., \( b_{i,k} = 1 \) for the valid sensor \( i \in S \) and \( b_{i,k} = 0 \) for the naive sensor. The information contributions for these naive sensor nodes are zero since the naive nodes have no valid measurement about current human target. The third step is to perform the hybrid consensus iteration for \( L \) steps. For each iteration \( l \), the \( i_{th} \) node sends its prior informations \( (y_{i,k}^{l-1}, Y_{i,k}^{l-1}) \), information contributions \( (u_{i,k}^{l-1}, U_{i,k}^{l-1}) \) and \( b_{i,k}^{l-1} \) to neighbor nodes and receives these consensus quantities from neighbors in parallel, then the consensus on this sensor node is performed according to (7), (8) and (9).

The Metropolis weights are used with the DHCF due to its faster convergence rate than the maximum-degree weights as shown in [24], which is defined as
\[
\epsilon_{i,j,k} = \begin{cases} 
1 - \max \{d_{i,k}, d_{j,k} \} & \text{if } j \in N_i \\
1 - \sum_{j \in N_i} \epsilon_{i,j,k} & \text{if } i = j \\
0 & \text{otherwise}
\end{cases}
\]
where \( d_{i,k} \) and \( d_{j,k} \) are the degrees of the camera node \( i \) and camera node \( j \) at the time step \( k \) respectively. Finally, the estimated human skeleton results of the algorithm at the discrete time step \( k \) are derived by (12). The DHCF algorithm requires no information of the number of sensor nodes of the camera network, which is suitable for scalable networks.

### B. Skeleton Based Human Action Recognition

To demonstrate the performance of the fusion algorithm, and to show potential application, we use a skeleton based human action recognition algorithm proposed by [25] to compare the recognition rate. Each skeleton joint can be treated as a point in the Lie group, and the 3D geometric relationships between skeleton joints can be modeled as rotations and translations. The human actions are then modeled as curves in the Lie group, which are mapped to vectors in their Lie algebra. The action classification can be further processed by a combination of dynamic time warping, Fourier temporal pyramid representation and linear SVM.

### III. Experimental Setup and Results

We construct a RGBD camera network using four Kinect V2 sensors (C1, C2, C3 and C4), which has a loop topology as shown in Fig. 1. The cameras are calibrated using a chessboard, followed by an iterative closest point (ICP) optimization algorithm on the point cloud captured by four Kinect V2 sensors to achieve more accurate calibration result. Each camera is connected to a computer to record and process the skeleton sequences.

Eight actors are invited to stand in the central position surrounding by cameras as shown in Fig. 1, and do 20 action classes according to the MSRAAction3D dataset [23], i.e., two hand wave, side-boxing, bend, forward kick, side kick, jogging, tennis swing, tennis serve, golf swing, push and throw, high arm wave, horizontal arm wave, hammer, hand catch, forward punch, high throw, draw x, draw tick, draw circle, hand clap. To show the occlusion problems, we ask each actor to do the same action class five times in three directions, i.e., one in middle, twice in left and twice in right, which are shown as red arrows in Fig. 1.

The skeletons are recorded using the Microsoft Kinect SDK library, which have poor quality when the camera has a side view or back view of the target human. Each skeleton has 20 joints, and each joint has a confidence score, which is corresponding to the probability of the joint detection. However, the skeleton detection algorithm of the Kinect only considers that the human is facing the sensor, such that the algorithm can make mistake if the human shows backside to the Kinect, i.e., the skeleton joint has a high confidence score for the wrong detection (right elbow labeled as left elbow, etc.). To address these problems, the joint angle and joint position constraints are used to correct the confidence score, i.e., a front view or a rear view about the human can be discriminated.
Fig. 2: The raw skeleton sequences of the high arm wave action captured from Kinect C1, where 3, 4, 6 and 8 are frame numbers. The blue color, red color and green color corresponding to torso part, right body part and left body part respectively. Since C1 is on the left side of the actor, the right arm of the actor is occluded by the torso, which result in incorrect estimation in the first two frames.

Fig. 3: The DHCF based skeleton fusion result of sensor node C1, which is converged after 9 iterations, where the joints with poor quality are corrected. Other nodes C2, C3 and C4 have the same estimation results as node C1, since the DHCF algorithm is converged.

We introduce the DHCF algorithm to fuse the skeleton from multiple views as shown in Algorithm 1, where the state transition matrix $F$ and Gaussian process noise $Q$ are defined as

$$
F = \begin{bmatrix}
1 & 0 & 0 & \Delta t & 0 & 0 \\
0 & 1 & 0 & 0 & \Delta t & 0 \\
0 & 0 & 1 & 0 & 0 & \Delta t \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
\end{bmatrix}, \quad (16)
$$

$$
Q = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 50 & 0 & 0 \\
0 & 0 & 0 & 50 & 0 & 0 \\
0 & 0 & 0 & 0 & 50 & 0 \\
\end{bmatrix}. \quad (17)
$$

The observation matrix $H$ and measurement noise $R$ are defined as

$$
H = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
\end{bmatrix}, \quad (18)
$$

$$
R = \begin{bmatrix}
20 & 0 & 0 & 0 & 0 & 0 \\
0 & 20 & 0 & 0 & 0 & 0 \\
0 & 0 & 20 & 0 & 0 & 0 \\
\end{bmatrix}. \quad (19)
$$

The iteration number $L$ can be set to different values, such that we can see the convergence rate of the algorithm.

For action recognition, the cross subject test setting is employed, i.e., we use half of the subjects (1, 3, 5, 7) for training and the other half (2, 4, 6, 8) for testing, since we have 8 individual persons to demonstrate the 20 action classes.

The average action recognition results are summarized in Table I. We can see that the DHCF based skeleton fusion algorithm has higher recognition rate than the results from

<table>
<thead>
<tr>
<th>Data source</th>
<th>Method</th>
<th>Recognition result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single view</td>
<td>Camera C1</td>
<td>82.75%</td>
</tr>
<tr>
<td></td>
<td>Camera C2</td>
<td>67.00%</td>
</tr>
<tr>
<td></td>
<td>Camera C3</td>
<td>75.50%</td>
</tr>
<tr>
<td></td>
<td>Camera C4</td>
<td>87.50%</td>
</tr>
<tr>
<td>Multiple view</td>
<td>UKF [13]</td>
<td>88.50%</td>
</tr>
<tr>
<td></td>
<td>IWCF (L=9) [19]</td>
<td>93.75%</td>
</tr>
<tr>
<td></td>
<td>DHCF (L=9)</td>
<td>94.50%</td>
</tr>
</tbody>
</table>
The recognition rates of DHCF based method outperform the others. It is clear that the DHCF based method outperforms the others.

The values in the confusion matrix present the recognition accuracy. It is clear that the DHCF based method outperforms the others. The confusion matrix of recognition results for all methods are shown in Fig. 4. which show that the DHCF has better performances than UKF and IWCF based methods.

The experiment result also shows that the DHCF based fusion algorithm outperforms the IWCF based one for skeleton fusion, since the DHCF has faster convergence rate and can preserve the consistency of the local filters, such that the novel information is never overestimated [22], [23]. Furthermore, we also compare our method with the state of the art work OpenPTrack [13], which is based on centralized unscented Kalman filters (UKF) to track multiple people with asynchronous data sources. For comparison, we remove the data association part of the original OpenPTrack algorithm, since our dataset only has one person for each image frame. We keep the asynchronous mechanism and use the same filter parameters as the original OpenPTrack algorithm. The results show that our method has better recognition accuracy than the UKF based OpenPTrack algorithm as shown in the Table I, i.e., 94.50% vs 88.50%. The confusion matrix of recognition results for all methods are shown in Fig. 4. which show that the DHCF has better performances than UKF and IWCF based methods.

IV. CONCLUSION AND FUTURE WORKS

In this work, we propose a dynamic hybrid consensus filter for skeleton fusion from multiple views, which can handle occlusion problem efficiently and further improve the action recognition accuracy. To demonstrate the idea, we use a distributed RGBD camera network and collect 20 action classes from 8 individual persons. Finally, a Lie group based action recognition method is used to show the improved performance of the proposed idea. In future, a distributed data association method can be combined with our current work to handle multiple human tracking and pose estimation problems.

REFERENCES


