MULTI-PEOPLE POSE TRACKING THROUGH VOXEL STREAMS

Masamichi Shimosaka, Yuichi Sagawa, Tomomasa Sato and Taketoshi Mori

The University of Tokyo

Email: {simosaka, sagawa, tmori}@ics.t.u-tokyo.ac.jp

ABSTRACT

Vision based human articulated body pose tracking has been historically important. Because analyzing multiple human activities, especially interaction between human in cluttered scenes is essential in visual surveillance scenarios, multiple people tracking has enjoyed much attention in human robot interaction research in recent years. In this paper, we newly introduce a robust framework for multiple people pose tracking. The notable aspects of our approach are real-time ensuring speed (up to 30 fps), flexibility towards various complex motions and environments. Our work is inspired by the success of multiple view approach, especially voxel based techniques. The use of voxel data leads to viewpoint-free estimation, which benefits in that reconstruction of a training model is needless in different multi-camera arrangements. We add simple tracking-based volume segmentation algorithm to retain practical superiority of voxel based approach. Furthermore, our framework successfully obtains multiple body pose estimation in real-time even when people contacts with each other occurs in the scene, which is not addressed in the conventional approaches. We demonstrate the effectiveness of our approach with experiments on indoor cluttered scene sequences.

Keywords— Multie-body tracking, Visual surveillance, Voxel intersection, approximated near neighbour search

1. INTRODUCTION

Vision based human articulated body pose tracking, also known as markerless motion capture in recent years, has been historically important in computer vision community because of its potential to many applications. For surveillance in intelligent environment (e.g. Aware Home [1]), people are often the main objects of interest and tracking body poses leads for inference of human activities. While there has been a long history of research in articulated tracking, body tracking of only single person in controlling environment is focused [2, 3, 4]. This is due to the difficulty of 3D articulated body tracking from the high-dimensional search spaces of body configuration with multi modal posterior distributions under cluttered scenes. Several state of the art techniques with bottom-up approach [5, 6] can easily get confused in these scenarios.

Because analyzing multiple human activities, especially interaction between human in cluttered scenes is essential in visual surveillance scenarios, multiple people tracking has enjoyed much attention in human robot interaction research in recent years. In spite of recent advances in multiple body tracking [7], difficulties in multiple body tracking leads manual initialization (partly solved by [8]), the view restriction [9], the lack of real-time operation [10], and requirements of target actions prior, which limits the variety of the target actions [11]. The notable difficulty of multiple human pose tracking arises from interactions of the target peoples where people overlap and partially occlude each other. In most recent advances in multiple body tracking [12], their body pose estimation is relatively simple thus the tracker fails when people contacts with each other.

1.1. Our approach

From a practical point of view on multiple body tracking in surveillance scenario: real-time operation, body contacts and accuracy, we leverage recent progress on multiple view approach, because multiple view approach reduces ambiguities in body pose estimation then obtains accurate results [13, 3]. Inspired by the success of multiple view, especially voxel based techniques [14, 15], this paper newly introduces a robust voxel based framework for multiple people pose tracking in real-time. The use of voxel data leads to viewpoint-free estimation, which benefits in that reconstruction of a training model is needless in different multi-camera arrangements. Based on this framework, we add simple tracking-based volume segmentation as preprocess of voxel-based trackers. This approach retains practical superiority of voxel based approach. Furthermore, our framework successfully obtains multiple body pose estimation in real-time even when people contacts with each other occurs in the scene, which is not addressed in the conventional approaches. This is our main contribution in this paper.

This paper makes the following contributions to ensure robust real-time multiple people pose tracking in surveillance scenario. We propose a combination of volume segmentation and fast single person body pose estimation from voxel streams. Although integrated approaches that simultaneously segments human region and estimates body has been proposed recently [16], it lacks real-time performance. As a part of the main contributions, we provide simple track-based volume labeling algorithm. To ensure real-time performance for estimating single person body poses after the voxel tracking, we leverage

978-1-4244-7493-6/10/$26.00

example-based approach [17]. This is one of the three major approaches for articulated body tracking[1].

The rest of the paper proceeds as follows. Related works are summarized in this section. Section II outlines example-based people pose tracking through voxel streams (EPTV) framework. Section III introduces a framework of multiple people pose tracking based on EPTV. Section IV presents results of several experiments about our framework. We conclude in section V with some directions for future research.

2. EXAMPLE-BASED POSE TRACKING FROM VOXELS: EPTV

2.1. Problem formulation

This section provides a brief overview of the example-based pose tracking from voxel streams (EPTV) [20]. In human pose estimation through voxel streams, 3D voxel data $\boldsymbol{v}(t)$, cube of the divided 3D space, is designed to be the input data. The 3D visual hull, an assembly of voxels, is reconstructed by volume intersection methods [21, 22]. In example-based approach, instead of outputting human body joint angle data $\boldsymbol{\theta}(t)$ in continuous quantity, output data $\boldsymbol{\theta}_{y(t)}$ is designed to be discrete. $y(t)$ indicates the estimated label of time t, which represents one of the N_y posture codebook $\boldsymbol{\theta}_j$ $(j = 1, \ldots, N_y) \equiv \boldsymbol{\mathcal{Y}}$ that are calculated beforehand. This configuration possibly causes a sparse inference result, which is an arguable point, but increase in posture candidates would lead to denseness of human posture state. Thus, example based inference can be considered as a sufficient approximation of continuous inference methods [17]. To make efficient codebook from motion capture databases, top-down or bottom-up approach of clustering [23] is often used. Empirical evaluation shows that over 300 thousands pose codebook ensures smooth pose tracking.

In example-based framework, inference is simplified to a comparison of likelihoods between 3D voxel data $\boldsymbol{v}(t)$ and human posture candidates $\boldsymbol{\mathcal{Y}}$ made from motion capture databases. For likelihood calculation, a feature vector $\boldsymbol{q}(t)$ is extracted from voxels $\boldsymbol{v}(t)$. On the other hand, posture candidate $\boldsymbol{\theta}_j \in \boldsymbol{\mathcal{Y}}$ is preprocessed into artificial voxel data $\boldsymbol{v}_j$, which enables feature vector $\boldsymbol{q}_j$ to be extracted. In this way, $\boldsymbol{q}(t)$ (query feature vector) and $\boldsymbol{q}_j$ $(j = 1, \ldots, N_y)$ (candidate feature vectors) are extracted. And then, likelihoods $\phi_j(t)$ $(j = 1, \ldots, N_y)$ between $\boldsymbol{v}(t)$ and $\boldsymbol{\mathcal{Y}}$ is evaluated through matching function S as $\phi_j(t) = S(\boldsymbol{q}(t), \boldsymbol{q}_j)$. Inference per frame is possible by selecting the label that outputs the maximum value among likelihoods $\phi_j(t)$ $(j = 1, \ldots, N_y)$ as

$$\hat{y}(t) = \mathrm{argmax}_j \phi_j(t). \tag{1}$$

In addition to the basic example-based approach, we use first order Markov property to smooth the output in this research. It is represented by a directed graphical model of motion. Motion sequences are scanned, and each frame is referenced with a posture label. During the scanning process, the transition frequency between previous label i and current label j is accumulated, and used to eventually derive the transition probability $T_{i,j}$. Although Taycher et al. [17] leverages maximum likelihood training framework to obtain the optimal transition probability, we use binary information for $T_{i,j}$. This is because the pose estimation performance does not get worse even if the matrix $T_{i,j}$ is represented as highly sparse matrix, thus we can reduce computational cost to calculate the likelihood. The graphical model of motion (see Figure 1) is preprocessed after constructing posture candidates, and used in the online inference process like Viterbi decoding [23]. The maximization of posterior probability, label $y(t)$ that outputs the maximum value among accumulated likelihoods $\{p_j(t)\}_{j=1}^{N_y}$ is calculated as

$$p_j(t) = \begin{cases} \phi_j(t) & (t = 0) \\ \max\limits_i (p_i(t-1)T_{i,j}) + \phi_j(t) & (t \neq 0) \end{cases} \tag{2}$$

$$y(t) = \mathrm{argmax}_j \, p_j(t). \tag{3}$$

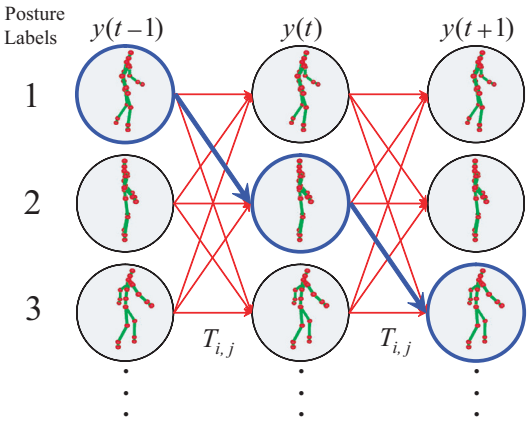

Fig. 1. Inference based on a Graphical Model of Motion

The estimation scheme mentioned above is summarized as Figure 2.

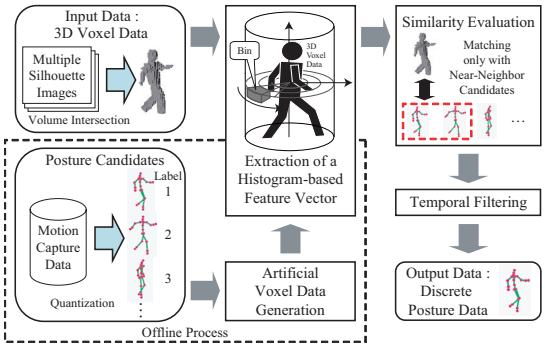

Fig. 2. Outline of the Human Pose Estimation Process

2.2. Keys to success in EPTV

The keys to success of this approach are 1) to make robust voxel features and 2) to reduce matching calculation $\phi_j(t)$. Representative candidates are extracted through a clustering process.

[1]The three contains example-based, generative [14, 18], and discrimnative [19, 4] approaches

As for feature extraction, we need view invariant feature representation such as 3D version of shape context, a natural extension of 2D silhouette shape descriptor [24]. We use a simple histogram-based feature called *cylindrical histogram feature* [20]. This will be explained in section IV.

Since the number of pose codebook is extremely large, it is impossible to maintain the real-time performance with the same estimation scheme. Hence approximated or fast near neighbor search approaches [25], such as Kd-tree or locality sensitive hashing [26], are needed. An efficient hashing framework called CSI-PSH [27], an extension of parameter sensitive hashing (PSH) [28] is employed in this research because of its simplicity and great performance.

3. MULTI-PEOPLE TRACKING ON EPTVS

3.1. Outline

In our framework, 3D voxel data is separated into multiple volumes through a volume labeling process, and then one EPTV is executed independently per person. The main advantage over multi-people tracking on monocular or stereo vision systems is that we can easily separate tasks of human segmentation and pose tracking. Flow of multiple people pose estimation is illustrated in Figure 3. To pursue multiple body pose estimation in real-time even when people contacts with each other occurs in the scene, which is not addressed in the conventional approaches, we must carefully design correct volume labeling algorithm. Key to success here is to use temporal information of voxel labels. This means that human ID annotated by volume labeling process is executed not independently in each frame but sequentially in successive frame.

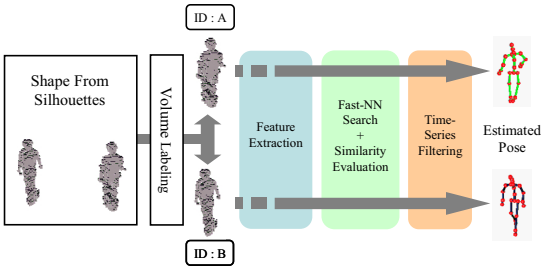

Fig. 3. Flow of Pose Estimation for Multiple People

3.2. Voxel segmentation through search type volume labeling

A normal 6-connectivity labeling (3D extension from 2D 4- or 8- neighbor morphological labeling) cannot deal with people connected to each other, because the volume will be assumed as a single volume. Therefore, we propose a new labeling process called *search type volume labeling*. This method can deal with people connected to each other by using time series information. The idea of *search type volume labeling* is to treat voxels independently so that it will not be affected by connectivity. The labeling procedures are summarized below.

1. In the detection stage, 6-connectivity labeling is applied. Volumes consisting of more than a certain number of voxels (depends on voxel size, about 1000 in our configuration) are assumed to be that of human.

2. After detection, each voxel is tracked by searching the minimum distance between the element of previously labeled volumes. A concept illustration in 2D is presented in Figure 4. The distance metric is defined as the *Manhattan distance* or the *city-block distance*. By introducing such a path-based distance metric and storing paths that have already been searched, repetition of search on the same path is avoided and leads to speed-up.

3. If the searched minimum distance of a voxel is lower than parameter T_v (allowance of distance), the voxel will succeed the label associated to the nearest element (voxel) of previously labeled volumes.

4. When number of voxels containing volume falls below a certain number, the person is assumed to be gone from the scene, and tracking is terminated.

This framework could not work appropriately if multiple people move along with contacting other people. However, this situation could be though as very rare in surveillance scenario. The proposed framework works sufficient in vast majority situation.

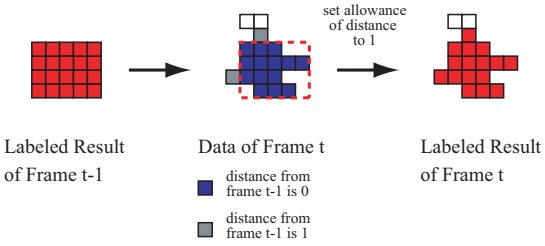

Fig. 4. Concept of labeling in 2D space

3.3. Mathematical formulation

This section formulates our framework. Let $\boldsymbol{v}(t)$ be voxels at time t. As a result of volume labeling, the I-th voxel $\{\boldsymbol{v}(t)\}_I$ in the collection is assigned with attribute as

$$\{\boldsymbol{v}(t)\}_I = \begin{cases} -1 & \text{:voxel doesn't exist} \\ 0 & \text{:voxel is invalid} \\ \text{positive value} & \text{:human ID} \end{cases} . \quad (4)$$

We use a distance function between I-th and $\tilde{I}$-th voxel with $D_v(I, \tilde{I})$. This function indicates a distance from I-th to $\tilde{I}$-th voxel via 6 connectivity voxel path. Attributes of $\{\boldsymbol{v}(t)\}_I$ are derived from the attributes of voxel collection $\boldsymbol{v}(t-1)$. To assign attribute of I-th voxel at time t, the framework retrieves $\acute{I}$-th voxel that minimizes $D_v(I, \acute{I})$ where $\{\boldsymbol{v}(t-1)\}_{\acute{I}} > 0$. Then the attribute of I-th voxel is updated as $\{\boldsymbol{v}(t)\}_I = \{\boldsymbol{v}(t-1)\}_{\acute{I}}$ if $D_v(I, \acute{I})$ is less than T_v. In our implementation, the

threshold T_v is set to 7. This update procedure for $\{\boldsymbol{v}(t)\}_I$ is summarized as

$$\{\boldsymbol{v}(t)\}_I = \begin{cases} -1 & \text{if voxel doesn't exist} \\ 0 & \text{if } D_v(I,\acute{I}) > T_v \\ \{\boldsymbol{v}(t-1)\}_{\acute{I}} & \text{if } D_v(I,\acute{I}) \leq T_v \end{cases} \quad (5)$$

$$\acute{I} = \underset{\tilde{I}\in\{\boldsymbol{v}(t-1)\}_{\tilde{I}}>0}{\operatorname{argmin}} D_v(I,\tilde{I}). \quad (6)$$

3.4. Online-processing by parallelization and pipeline

The overall process works in a parallel calculation framework with multiple computers to remain real-time operation. This framework is based on a 3 stage pipeline process as shown in Figure 5. Stage 1 is executed on server machines where phases independent to cameras correspond. On the other hand, stage 2 and 3 are executed on the client machine and respectively correspond to reconstruction of 3D voxel data and the main human pose tracking phase. In this framework, a delay of 2 frames will occur, but 30 fps online processing will be possible if every stage completes within 33 msec.

As a matter of course, total computational cost in pose tracking is proportional to the number of the target peoples. On the other hand, the tasks are independent to each other after multiple volumes are acquired, thus the pose estimation task could be fully parallelized. We parallelize these tasks by requesting computation on intermediate servers, so that processing time will not be proportional to the number of people. In our hardware configuration, 2 additional servers are operating, so the system can deal with 3 people in maximum (the client machine can also execute the pose estimation task).

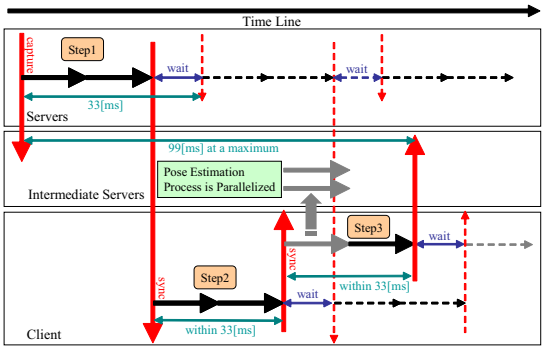

Fig. 5. Parallel Calculation Framework

4. EXPERIMENTAL RESULTS

4.1. Implementation issues

We deploy our framework in daily life simulated space. In our implementation, 8 cameras manufactured by Point Grey research are fixed to the ceiling and are used to synchronously capture multiple VGA images. The images are captured at 30 fps. Each camera is connected to single standard PC with Intel Core duo micro architectures via IEEE 1394. We leverage 2 standard PCs as intermediate servers for multiple people pose tracker.

4.2. Preprocessing

This section describes features used in this experiment and their quantity. The volume intersection method [21, 22] reconstructs 3D object shape from multiple silhouette images. The silhouette images are obtained from simple background subtraction method, such as Otsu method or Gaussian mixture pixel intensity models. 3D shape will consist of an assembly of voxels, which represent cubes of the divided 3D space. 8 cameras are used to capture multiple VGA images. Voxel size is set to 35mm, and 3D space is divided into a 120×120×68 resolution.

As a query vector calculated from voxels, we use simple histogram-based feature called *cylindrical histogram feature* [20] (see Figure 2). In this feature, 3D space is divided into multiple bins based on a central axis set to the center of gravity of voxel data (3D space is divided in angle, height, and radius directions). Voxels are voted to the corresponding bins, and the resulting histogram will be normalized. Resolution of angle, height, and radius directions are set to 18, 8, and 3. Thus the dimension of the feature vector became 432.

For pose codebook construction, motion capture data downloaded from [29] are used (motion data include large body rotations, complex motions, and self occlusions), and resulted in a total of 322,992 posture codebook. We use a kind of agglomerative hierarchical clustering algorithm. This makes it possible to acquire a more uniform and dense distribution in human posture space. In our evaluation, likelihood is based on the Bhattacharyya coefficient [30] as $\phi_j(t) = \sum_{r=1}^{r_q} \sqrt{\{\boldsymbol{q}(t)\}_r \{\boldsymbol{q}_j\}_r}$.

For $V(\boldsymbol{\theta})$, artificial voxel data is generated by approximating body links as a cylinder or an ellipsoid (approximated radii are configured manually). At first, joint angle format data is reflected to the human model. Then, voxels included in the approximated region of each body link is extracted to configure a voxel data.

For making CSI-PSH, we use 579,731 similar pairs with 174,937 dissimilar pairs (total of 754,668 pairs) for training pairs. Additionally, 233,062 elements of evaluation data were prepared for feedback training and hash functions were trained under these conditions. The other variables in CSI-PSH are empirically set to use [27] as reference in this experiment.

4.3. Tracking results

Experimental Results are presented in Figure 6. This implies that our system successfully capture various complex motion. Table 1 shows that overall processing time. Each stage has completed within 33ms, which signified that online processing of 30fps has achieved (however, certain amount of delay will occur). Quantitative performance evaluation shows that the positional error at human hand of our pose tracker is about 100 ∼ 250mm and angular error are less than 7 degree. Note that the height of one of the tracked people differs from the height of the others (1.80 m and 1.60 m).

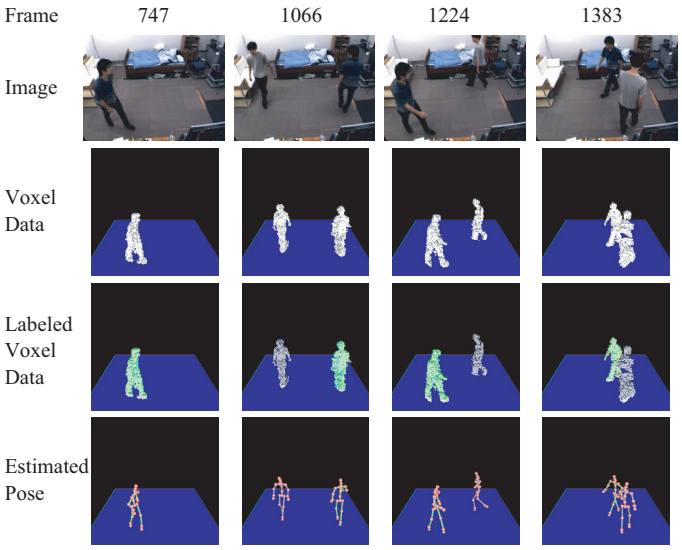

Fig. 6. Experimental result of human pose estimation (best viewed in color)

Table 1. Processing Time

# of peoples	1	2 (sequential)	2 (parallel)
Background subtraction	22 msec	24 msec	24 msec
Volume intersection	2 msec	3 msec	3 msec
Summary of 1st step	24 msec	27 msec	27 msec
Volume integration	4 msec	8 msec	8 msec
Labeling	2 msec	4 msec	4 msec
Summary of 2nd step	6 msec	12 msec	12 msec
Pose estimation	**20 msec**	**40 msec**	**20 msec**
Rendering	5 msec	10 msec	10 msec
Summary of 3rd step	**25 msec**	**50 msec**	**30 msec**

4.4. Result for volume labeling under people collision

We observed the result of our framework under multiple people collision to validate the search type volume labeling. The result of the sequence under people collision occurrence is shown in Figure 7. The different color in voxel data indicates different human ID. In the figure, result with naive 6-connectivity volume labeling is shown as "Labeling Result1" where human ID is not correctly assigned. In contrast, search type volume labeling (Labeling Result2) succeeds in human ID annotation even when a person contacts each other.

Though the framework described in section III avoids fatal error when people contact each other, partial error is inevitable after the contact (see Labeling Result2). To avoid these errors, we added outlier elimination process. In this modification, I-th voxel is updated as $\{\boldsymbol{v}(t)\}_I = 0$ when it is classified outlier. The final result of our framework with outlier elimination is shown in "Labeling result3".

The proposed volume labeling method requires $3 \sim 5$ msec under the situation where two people occur in the environment with $|\boldsymbol{v}(t)| \approx 2000 \sim 5000$, while naive 6-connectivity labeling needs $2 \sim 4$ msec.

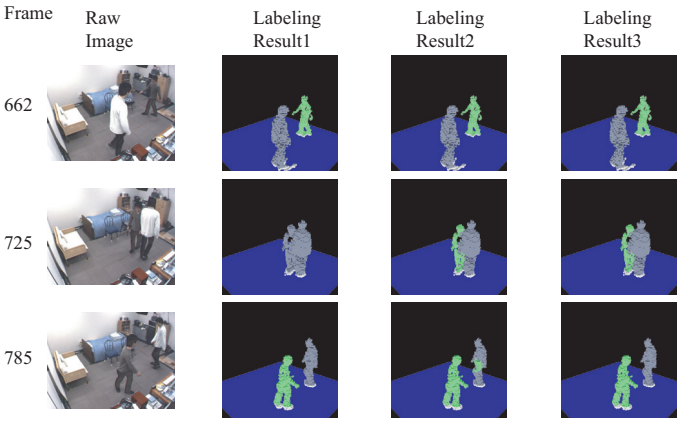

Fig. 7. Volume labeling under people collision occurrence

4.5. Activity monitoring with multi-people pose tracking

Figure 8 shows that our tracking system provides functionality of multi peoples interaction under daily house environment. In the start of this scenario, one person is sitting on a chair to watch TV program. In a moment, the other resident entered the room then sits on the chair besides the first resident. Finally they left after watching TV program. Though the maximum number of peoples tracked in our framework highly depends on the way of camera arrangement, our empirical result shows that our approach is able to track within 4 peoples simultaneously under this experimental setting.

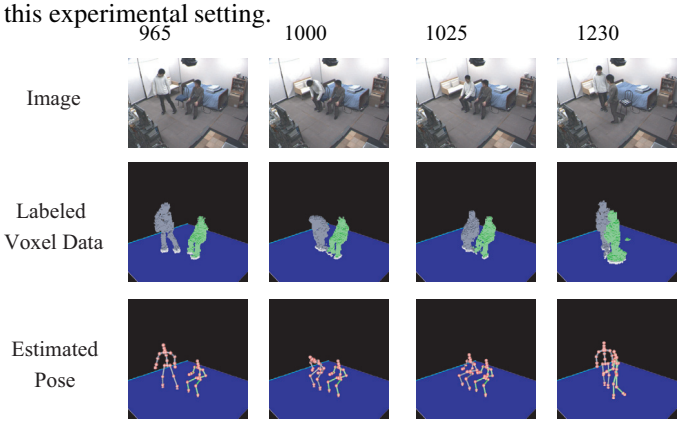

Fig. 8. Experimental results on a TV Watching situation

5. CONCLUSION

A novel approach to recover multiple human pose through voxel streams has been proposed. Experimental results show that real-time processing up to 30 fps has been achieved by introducing simple volume labeling and example-based pose tracking. Future tasks are to realize not only pose but also shape reconstruction under clothing [31] and to tackle the occlusion problem by furniture for practical daily house use.

6. REFERENCES

[1] C. Kidd, R. Orr, G. Abowd, C. Atkeson, I. Essa, B. MacIntyre, E. Mynatt, T. Starner, and W. Newstetter, "The

Aware Home: A living laboratory for ubiquitous computing research," in *CoBuild 1999*, pp. 191–198.

[2] J. Deutscher, A. Blake, and I. Reid, "Articulated body motion capture by annealed particle filtering," in *CVPR 2000*, pp. II–2126–2134.

[3] L. Ren, G. Shakhnarovich, J. Hodgins, H. Pfister, and P. Viola, "Learning silhouette features for control of human motion," *ACM Trans. on Graphics*, vol. 24, no. 4, pp. 1303–1331, 2005.

[4] C. Sminchisescu, A. Kanaujia, Z. Li, and D. Metaxas, "Discriminative density propagation for 3D human motion estimation," in *CVPR 2005*, pp. I–390–397.

[5] X. Ren, A. Berg, and J. Malik, "Recovering human body configurations using pairwise constraints between parts," in *ICCV 2005*, pp. I–824–I–831.

[6] D. Ramanan and D. Forsyth, "Finding and tracking people from the bottom up," in *CVPR 2003*, pp. II–467–II–474.

[7] J. Mitchelson and A. Hilton, "Simultaneous pose estimation of multiple people using multiple-view cues with hierarchical sampling," in *BMVC 2003*.

[8] D. Ramanan, D. Forsyth, and A. Zisserman, "Strike a pose: Tracking people by finding stylized poses," in *CVPR 2005*, pp. I–271–278.

[9] M. Andriluka, S. Roth, and B. Schiele, "People-Tracking-by-Detection and People-Detection-by-Tracking," in *CVPR 2008*.

[10] M. Lee and R. Nevatia, "Human pose tracking using multi-level structured models," in *ECCV 2006*, pp. III–368–381.

[11] S. Gammeter, A. Ess, T. Jäggli, K. Schindler, B. Leibe, and L. V. Gool, "Articulated multi-body tracking under egomotion," in *ECCV 2008*, pp. II–816–830.

[12] C. Canton-Ferrer, J. Salvador, J. Casas, and M.Pardás, "Multi-person tracking strategies based on voxel analysis," in *Proc. of the International Evaluation Workshops CLEAR 2007 and RT 2007*, pp. 91–103.

[13] L. Sigal, S. Bhatia, S. Roth, M. Black, and M. Isard, "Tracking loose-limbed people," in *CVPR 2004*, pp. I–421–428.

[14] I. Mikic, M. Trivedi, E. Hunter, and P. Cosman, "Human body model acquisition and tracking using voxel data," *International Journal of Computer Vision*, vol. 53, no. 3, pp. 199–223, 2003.

[15] Y. Sun, M. Bray, A. Thayananthan, B. Yuan, and P. Torr, "Regression-based human motion capture from voxel data," in *BMVC 2006*, pp. 277–286.

[16] M. Bray, P. Kohli, and P. Torr, "PoseCut: Simultaneous segmentation and 3D pose estimation of humans using dynamic graph-cuts," in *ECCV 2006*, pp. II–642–655.

[17] L. Taycher, G. Shakhnarovich, D. Demirdjian, and T. Darrell, "Conditional random people: Tracking humans with CRFs and grid filters," in *CVPR 2006*, pp. I–222–229.

[18] K. Ogawara, X. Li, and K. Ikeuchi, "Marker-less human motion estimation using articulated deformable model," in *ICRA 2007*, pp. 46–51.

[19] A. Agarwal and B. Triggs, "3D human pose from silhouettes by relevance vector regression," in *CVPR 2004*, pp. II–882–888.

[20] Y. Sagawa, M. Shimosaka, T. Mori, and T. Sato, "Fast online human pose estimation via 3D voxel data," in *IROS 2007*, pp. 1034–1040.

[21] W. Martin and J. Aggarwal, "Volumetric description of objects from multiple views," *IEEE Trans. on PAMI*, vol. 5, no. 2, pp. 150–158, 1983.

[22] A. Laurentini, "How far 3D shapes can be understood from 2D silhouettes," *IEEE Trans. on PAMI*, vol. 17, no. 2, pp. 188–195, 1995.

[23] D. MacKay, *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2005.

[24] S. Belongie, J. Malik, and J. Puzicha, "Shape context: A new descriptor for shape matching and object recognition," in *Advances in Neural Information Processing Systems 13*, 2001, pp. 831–837.

[25] G. Shakhnarovich, T. Darrell, and P. Indyk, *Nearest-Neighbor Methods in Learning and Vision: Theory and Practice*. The MIT Press, 2006.

[26] A. Gionis, P. Indyk, and R. Motwani, "Similarity search in high dimensions via hashing," in *VLDB 1999*, pp. 518–529.

[27] M. Shimosaka, Y. Sagawa, T. Mori, and T. Sato, "3D voxel based online human pose estimation via robust and efficient hashing," in *ICRA 2009*, 2009, pp. 3577–3582.

[28] G. Shakhnarovich, P. Viola, and T. Darrell, "Fast pose estimation with parameter sensitive hashing," in *ICCV 2003*, pp. II–750–757.

[29] "www.mocapdata.com."

[30] T. Kailath, "The divergence and Bhattacharyya distance measures in signal selection," *IEEE Trans. on Comm. Technology*, vol. 15, pp. 52–60, 1967.

[31] A. Balan and M. Black, "The naked truth: Estimating body shape under clothing," in *ECCV 2008*, pp. II–15–29.