2008 IEEE/RSJ International Conference on Intelligent Robots and Systems
Acropolis Convention Center
Nice, France, Sept, 22-26, 2008

Learning Meaningful Interactions from Repetitious Motion Patterns

Koichi Ogawara and Yasufumi Tanabe and Ryo Kurazume and Tsutomu Hasegawa

***Abstract*— In this paper, we propose a method for estimating meaningful actions from long-term observation of everyday manipulation tasks without prior knowledge as part of an action understanding framework for life support robotic systems. The target task is defined as a sequence of interactions between objects. An interaction that appears many times is assumed to be meaningful and repetitious relative motion patterns are detected from trajectories of multiple objects. The main contribution is that the problem is formulated as a combinatorial optimization problem with two parameters, target object labels and correspondences on similar motion patterns, and is solved using local and global Dynamic Programming (DP) in polynomial time $O(N \log N)$, where N is a total amount of data. The proposed method is evaluated against manipulation tasks using everyday objects such as a cup and a tea-pot.**

I. INTRODUCTION

The applicable areas of robotics technology have been rapidly expanding and "supporting our life in everyday environment" is becoming one of the key applications.

To determine what robotic behavior is appropriate in a certain situation, the system should understand what a human is doing at the moment. A common approach to understand human behavior is to design a set of necessary and sufficient task-dependent recognizers that detect significant actions and capture the necessary parameters to describe the tasks [1], [2], [3], [4], [5]. Various tasks are tackled in this approach: including assembly planning [1], [2], soft object manipulation [3], whole body motion generation [4], [5], etc. However, unlike in the well-organized environment such as a factory, the variety of human behavior in everyday environment is infinite, thus it is not practical to prepare recognizers to cover all the possible daily activities.

For this reason, the desirable system should have a mechanism to obtain and accumulate new knowledge, i.e. new causal relationship, incrementally from observation. As a bootstrap process to realize this mechanism, we are interested in finding structured information in observations that can be extracted without prior knowledge about the presented task. The basic idea is if a particular motion pattern appears many times in observations, this pattern must be meaningful to the demonstrator or to the task. When the system detects some repetitious motion patterns such as preparation for breakfast or reading a news paper while observing daily activities for a long period time, e.g., several days, these patterns are marked as meaningful actions and the causal relationship between meaningful actions can be used to predict the next action of a user that can be supported in some ways by the robotic system, e.g., bringing a news paper.

K. Ogawara is with Faculty of Engineering, Kyushu University, Fukuoka, JAPAN ogawara@is.kyushu-u.ac.jp

Y. Tanabe is with Department of Electrical Engineering and Computer Science, Kyushu University, Fukuoka, JAPAN

R. Kurazume and T. Hasegawa are with Faculty of Information Science and Electrical Engineering, Kyushu University, Fukuoka, JAPAN

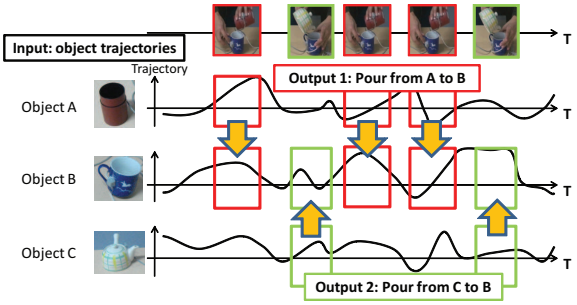

Fig. 1. Target Problem

In the scope of this paper, we assume that daily activities are composed of a sequence of interactions between two manipulated objects, and we propose a method for detecting repetitious interactions observed from motion trajectories of multiple environmental objects.

The problem is formulated as a combinatorial optimization problem with two parameters, target object labels and correspondences on similar motions, and is solved using local and global Dynamic Programming (DP) in polynomial time.

The contribution of the proposed method is two fold. The first contribution is to find repetitious motion patterns in temporal data streams without a reference pattern in polynomial time $O(N \log N)$, where N is a total amount of data, while most of the previous methods require $O(N^2)$ time or a reference pattern which is not practical in our applications where we have to deal with long-term observation without prior knowledge about the task. The second contribution is that it can estimate which two objects among many candidates constitute each meaningful interaction, while most of the previous methods assume a single object or a single trajectory.

This paper is organized as follows. In Section II, the target problem is explained and the related research as well as the overview of the proposed algorithm is presented. In Section III, the details of the proposed method is described. The experimental results and evaluation are shown in Section IV and we conclude the paper in Section V.

II. LEARNING MEANINGFUL INTERACTIONS

In this section, the problem of finding meaningful interactions between two objects from trajectories of multiple objects is examined. Here, we assume that an interaction that

appears many times in observations is meaningful, thus the problem becomes to find repetitious relative motion patterns between two particular objects.

Fig.1 shows an example of the target problem. There are three objects in the scene and the input to the algorithm is the three trajectories corresponding to each object. In this case, the desired output would be the two types of interactions: (1) "Pour from A to B" which appears three times and (2) "Pour from C to B" which appears two times. In both interactions, a set of segments are detected in which all the relative trajectories are quite similar to each other.

A. Related Research

There are roughly two approaches to detect repetitious motion patterns. One is a recognition approach where repetitious motion patterns are spotted by a set of recognizers. The other is a pattern matching approach where Dynamic Programming is typically used to detect similar patterns.

1) Recognition Approach: Mori et al. proposed a hierarchical recognition framework where multiple HMM-based recognizers for different action types and different abstraction levels run in parallel and inconsistencies in the result are resolved by the relationship in the action hierarchy[6]. However, the labelled training data must be provided for the recognizers and it cannot deal with new actions.

Zhao et al. proposed a structured representation of the motion primitives that satisfies MDL and recognized ballet sequence[7]. However, the typical motion patterns encoded in each recognizer are quite short and it is difficult to spot a long and complex interactions between objects in this approach.

2) Pattern Matching Approach: Bobic et al. represented a set of training trajectories of gestures as a sequence of states and recognized an input gesture using Dynamic Programming under a state transition framework[8]. However, this method requires manually segmented and labelled training data for each gesture.

To relax the requirement for pre-segmentation, Ogawara et al. proposed a method for detecting a set of repetitious motion patterns from multiple observations of the same task using multi dimensional Dynamic Programming[9]. However, multiple demonstrations should be provided in that the same types of motion patterns appears in the same order.

To overcome this problem, Uchida et al. proposed a method for detecting a set of repetitious motion patterns from a single observation using logical DP matching[10]. However, the computational complexity is $O(N^2)$ and it cannot deal with multiple objects.

Ogawara proposed a method for detecting a set of repetitious motion patterns from a single observation[11]. It can deal with multiple objects, however it solves a combinatorial optimization problem by a stochastic method, Markov-Chain Monte Calro (MCMC), and the computational efficiency becomes dramatically degraded as the amount of data increases.

B. Proposed Method Overview

In this paper, a deterministic method for detecting repetitious motion patterns from a single observation is proposed.

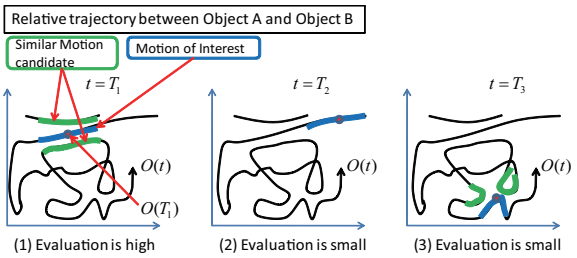

Fig. 2. Evaluation of Repetitious Motion Pattern. There is a single trajectory, but some parts are omitted to better visualize it.

Fig.2 shows the 2D slice of a relative trajectory $O(t)$ between two objects. If a motion pattern of interest at around time t has the similar motion patterns at other time frames on the same trajectory, this pattern can be considered as meaningful since this particular pattern appear multiple times. Because we take a relative trajectory between objects as input, the similarity can be evaluated by the degree of overlap between their trajectories in the eucledian space.

Fig.2 shows the three typical cases. In case (1), the motion pattern of interest at around $t = T_1$ has multiple similar motion pattern candidates in the neighborhood, thus the evaluation that the interaction between object A and B at around t is meaningful becomes high. In case (2), there is no similar motion pattern candidates at around $t = T_2$, thus the evaluation becomes small. In case (3), there are multiple motion pattern candidates in the neighborhood at around $t = T_3$, however the shape of them are completely different, thus the evaluation also becomes small.

The proposed algorithm evaluates the above mentioned criteria for each data point along the entire relative trajectory and integrate the evaluation in a globally consistent way.

The computational complexity of the proposed method is $O(N \log N)$, where N is a total amount of data. This is achieved by efficiently constraining the search space for finding the similar motion patterns within the neighborhood of each data point using kd-tree search algorithm.

Also, the proposed method can estimate which two objects among multiple object candidates constitute a repetitious motion pattern of interest. This is achieved by formulating the problem as a label assignment problem. This combinatorial optimization problem can be solved by Dynamic Programming which naturally solves the determination problem of the target object in the scene.

III. Detection of Repetitious Motion Patterns

A. Problem Formulation

The problem is formulated as a combinatorial optimization problem regarding to two parameters X, Y. Given observation O, these two parameters are solved for each object Obj_m in the scene using Maximum-A-Posteriori(MAP) estimate of the probability function $P(X, Y|O)$. Fig.3 shows the overview of the formulation.

The target object label $X = \{\bigcup x_t | x_t \in \{A, B, \cdots, N\}\}$ indicates the target object in the scene with which Obj_m

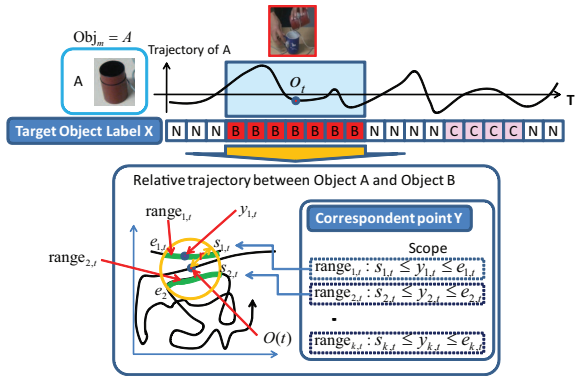

Fig. 3. Label Assignment Problem with Hidden Parameters

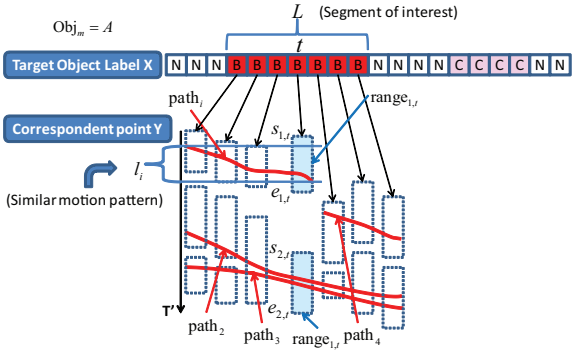

Fig. 4. Estimation of Correspondences Y using Local DP Matching

interacts at time t. If Obj$_m$ does not interact with any object at time t, label N is assigned to x_t.

The correspondence parameter $Y = \{\bigcup y_{k,t} | s_{k,t} \le y_{k,t} \le e_{k,t}\}\}$ indicates the time frame whose data point $O(y_{k,t})$ lies on a similar motion pattern and corresponds to the data point $O(t)$ at time t. Fig.3 shows an typical example in the middle of estimation process where target label B is assigned to x_t and the relative trajectory between object Obj$_m = A$ and B is of interest at around time t. The range of $y_{k,t}$ is limited within range$_{k,t}$($s_{k,t} \le y_{k,t} \le e_{k,t}$) which is calculated beforehand as described in Section III-B.

The MAP estimation of the probability function $P(X, Y|O)$ is re-written as in eq.(1) using Bayes's theorem.

$$\underset{X,Y}{\operatorname{argmax}} P(X, Y|O) \propto \underset{X,Y}{\operatorname{argmax}} P(O|X, Y)P(Y|X)P(X) \tag{1}$$

$P(O|X, Y)$ is an observation likelihood term given X and Y. $P(Y|X)$ is a frequency term which favors large number of similar motion patterns. $P(X)$ is a prior term, or a smoothness term, which penalizes the difference between consecutive target labels.

To realize MAP estimation, we employ an iterative framework as follows.

1) Initialization
2) Estimation of Y using local DP matching (X is fixed)
3) Estimation of X using global DP (Y is fixed)
4) Go to 2) until convergence

In the remainder of this section, the above mentioned procedure is explained in details.

B. Initialization

As mentioned in the previous section, range$_{k,t}$ (the range of $y_{k,t}$) is calculated as follows.

First of all, Kd-tree, a binary tree, is constructed on observed 6 D.O.F. relative trajectory space: 3 D.O.F for relative position and 3 D.O.F. for relative orientation. The entire observation data is recursively divided by a hyper-plane perpendicular to the axis along which the variance of the remaining data points becomes maximum. One Kd-tree is built for each relative trajectory. The computational cost for Kd-tree construction is $O(N \log N)$.

To determine range$_{k,t}$, Kd-tree is searched and the set of data points C_t whose distance to $O(t)$ is less than r is found as in eq.(2).

$$C_t = \{\bigcup O(j) | \parallel O(j) - O(t) \parallel < r, 0 \le j \le T\} \tag{2}$$

Then, the data points in C_t are sorted along time domain and the consecutive data points are grouped so as to define the range$_{k,t}$, from $s_{k,t}$ to $e_{k,t}$, of the correspondent point to $O(t)$. As for the case in Fig.3, two range are found.

This process is applied to all the data points $O(t)\{1 \le t \le T\}$ and range$_{k,t}$ is determined for all the time frame t. The total computational cost is $O(N \log N)$. This cost is the bottle neck of the proposed algorithm.

Lastly, X is initialized so that no-interaction label N is assigned to all x_t.

C. Estimation of Y using Local DP Matching

1) Estimation of Initial Y using Local DP Matching: Given target object label X, X can be divided into segments where all x_t in each segment have the same label. Fig.4 shows an example where label B happens to be assigned to the segment at around t during the estimation process.

Here, we try to find similar motion patterns by estimating the correspondences Y on them using local DP matching.

As shown in Fig.4, we already have range$_{k,t}$ on this segment. The range of the correspondent time frame $y_{k,t}$ is limited within range$_{k,t}$. This means, the similar motion patterns must go through each of range$_{k,t}$ as shown in red lines in the figure.

Under this constraints, the optimal paths are calculated using Dynamic Programming. Here, we define the allowable step size in DP ranges from 0 to 2, so that l_i takes value between 0 and $2L$. Eq.(3) shows the recurrence equation of DP.

$$g(t, t') = s(t, t') + \min \begin{pmatrix} g(t-1, t'-2) \\ g(t-1, t'-1) \\ g(t-1, t') \end{pmatrix} \tag{3}$$

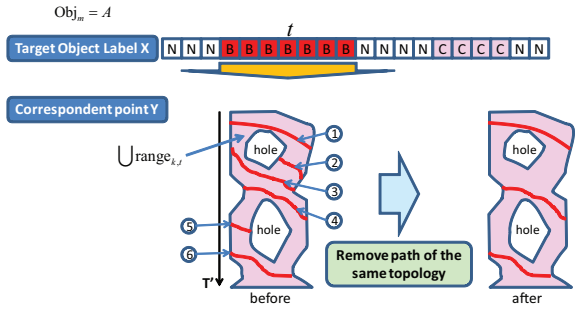

Fig. 5. Topological Reduction of Paths

where $s(t, t')$ means the similarity in pose, i.e. 3 D.O.F position and 3 D.O.F. orientation, at t and at t' on the relative trajectory between Obj$_m$ and Object B. $g(t, t')$ means accumulated cost along the path.

For each range$_{k,t}$, the longest and minimum cost path is calculated using Dynamic Programming where the cost is accumulated from the both terminal to range$_{k,t}$. The total number of calculated path is equal to the number of range$_{k,t}$ included in that segment. However, most of the paths are identical to each other in general and four distinct paths are finally found in the case of Fig.4.

Then, two parameters are calculated for each detected path$_i$. One is the ratio between the length of the segment at time t and the length of the corresponding similar motion pattern as shown in Fig.4 and is defined as in eq.(4). The other is the similarity between the segment and the similar motion pattern and is defined as a function of the average cost of DP matching dp_cost$_i$ along path$_i$ and is defined as in eq.(5).

$$l_i = \begin{cases} \frac{d_i}{L_i} (d_i \leq L_i) \\ \frac{L_i}{d_i} (L_i \leq d_i) \end{cases} \tag{4}$$

$$sim_{t,i} = \exp(-\beta \cdot dp\_cost_i) \tag{5}$$

2) Topological Reduction: Most of the detected paths in the previous section are redundant as shown in Fig.5. There are six paths found in "before" region that is the direct result of the local DP matching. Also, we can see that there are two holes in the region. This situation typically occurs when a user presents a similar motion several, in this case three, times without sufficient intervals.

Among six paths, the 2nd and 5th path are not important because they are apparently shorter compared with the 4th or 6th path. And the 3rd and 4th path are redundant and one of them should be selected.

Thus, we apply the following two operations to the candidates.

1) If a path starts or ends at a hole, remove it
2) If two paths can be completely overlap without going over the holes, remove one with lower evaluation

Then, we can get the non-redundant paths as shown in "after" region in Fig.5.

3) Definition of Probability Function: Finally, the probability function $P(O|X, Y)$, $P(Y|X)$ and $P(X)$ are defined as in eq.(6), eq.(9) and eq.(11).

$$P(O|X, Y) = \Pi P(o_t|x_t) P(O|x_t, y_t) \tag{6}$$

$P(o_t|x_t)$ means a velocity term that penalizes a static object and is defined as in eq.(7).

$$P(o_t|x_t) = 1 - \exp(-\alpha \cdot \text{velocity}_t) \tag{7}$$

where velocity$_t$ is the velocity of Obj$_m$ at time t.

$P(O|x_t, y_t)$ means a similarity term that penalizes different motion patterns and is defined as in eq.(8).

$$P(O|x_t, y_t) = \frac{\Sigma l_i \exp(-\beta \cdot \text{similarity}_i)}{\Sigma l_i} \tag{8}$$

where l_i and similarity$_i$ are defined in the previous section III-C.1.

$$P(Y|X) = \Pi P(y_t|x_t) \tag{9}$$

$P(y_t|x_t)$ means a frequency term that penalizes if the number of similar motion patterns is small and is defined as in eq.(10).

$$P(y_t|x_t) = 1 - \exp(-\gamma \cdot \Sigma l_i) \tag{10}$$

$$P(x) = \Pi P(x_t, x_{t+1}) \tag{11}$$

$P(x_t, x_{t+1})$ is a smoothing term that penalizes the difference between consecutive target labels and is defined as in eq.(12).

$$P(x_t, x_{t+1}) = T(x_t \neq x_{t+1}) \cdot K \tag{12}$$

where K is a constant and $T(s) = 1$ iff $s = \text{true}, T(s) = 0$ otherwise.

Lastly, the probability function where $x_t = N$ is defined as follows.

$$P(O|x_t = N, y_t) = \frac{\Sigma P(O|x_t \neq N, y_t)}{\#P(O|x_t \neq N, y_t)} \cdot \text{NL} \tag{13}$$

$$P(y_t|x_t = N) = \frac{\Sigma P(y_t|x_t \neq N)}{\#P(y_t|x_t \neq N)} \cdot \text{NL} \tag{14}$$

where NL is noise level.

D. Estimation of X using Global DP

Target Label X can be solved analytically via global DP matching.

As shown in Fig.6, a directional graph is constructed. $-logP(O|x_t, y_t) - logP(y_t|x_t)$ is assigned as a node weight and $-logP(x_t, x_{t+1})$ is assigned as an arc weight. Then, the minimum cost path is calculated in Dynamic Programming manner and the result represents the target object label X.

The computational cost of this process is $O(NM)$ where M is the number of objects in the scene.

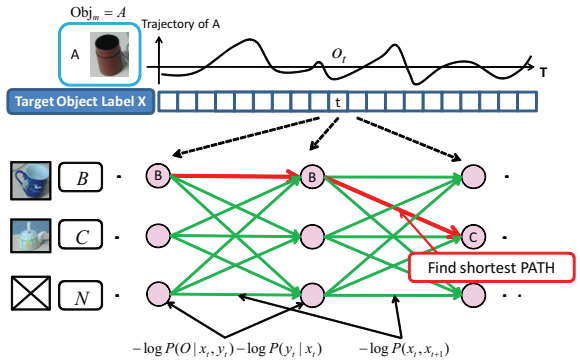

Fig. 6. Estimation of X using Global DP

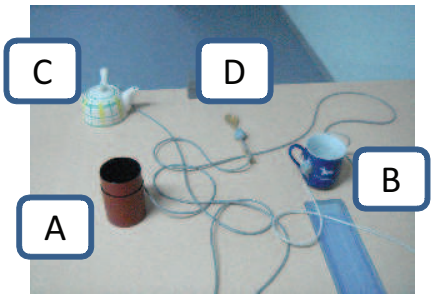

Fig. 7. Four Object used in the Experiment

I: Pour from A to B
II: move A to B
III: Spoon up with D in A
IV: Put material into B with D
V: Mix inside B with D
VI: Pour from C to B

Fig. 8. Six primitive Actions in the Scenario

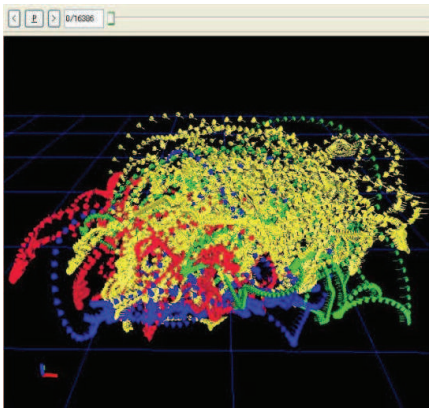

Fig. 9. Visualized Input Trajectory in Data-set 4 [16386 frames]

IV. Experimental Result

A. Experimental Setup

Four different manipulation tasks were performed by a subject and were used to evaluate the proposed method. There were four objects in the scene and an electromagnetic motion tracking system (Polhemus FASTRAK) was used to observe the trajectory of each object during demonstrations. Fig. 7 shows the objects used in this experiment.

As shown in Fig.8, six actions are defined. A subject was instructed to perform a task along a scenario made by combining six actions. To emulate the change in the environment during long-term observation, the subject was instructed to relocate the objects on the table several times so that the relative relationship between static objects was changed.

Fig.9 shows one of the visualized trajectory data set.

B. Evaluation

The proposed method is compared with the method proposed in [11]. In [11], the combinatorial optimization problem is solved by efficiently sampling the solution space via Markov Chain Monte Calro (MCMC) algorithm.

TableI, II, III, IV shows the presented scenario and the result of detecting repetitious motion patterns using two different methods.

As an error measure, Precision and Recall are calculated from True Positive (TP), False Positive (FP) and False Negative (FN) as defined in eq.(15).

$$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}}, \quad \text{Recall} = \frac{\text{TP}}{\text{TP}+\text{FN}} \tag{15}$$

C. Discussion

Since the data set used in the experiment were relatively short, a clear difference was not found between two methods. However, there was a tendency that the proposed method results in the lower False Positive rate.

The MCMC method becomes drastically slow if the amount of data increases. Even if the amount of data is moderate, it is sometimes difficult to decide when we should end the algorithm. On the other hand, the proposed method outputs the result in deterministic way and we do not have to worry about the termination of the algorithm.

Several constant parameters in the probability function are defined ad-hoc and they should be adjusted to achieve better error ratio.

V. Conclusion

This paper presents a method for detecting repetitious relative motion patterns among multiple trajectories in polynomial time.

The problem is formulated as a combinatorial optimization problem with two parameters and is solved using local

TABLE I

EVALUATION FOR DATA-SET 1 [1788 FRAMES]

Action	III	IV	V					
Number	3	0	0	TP	FN	FP	Precision	Recall
MCMC Method [11]	0	0	0	0	9	6	0.0	0.0
Proposed Method	0	0	1	1	8	0	1.00	0.11

TABLE II

EVALUATION FOR DATA-SET 2 [3819 FRAMES]

Action	I	V	VI					
Number	6	6	6	TP	FN	FP	Precision	Recall
MCMC Method [11]	5	6	6	17	1	8	0.68	0.94
Proposed Method	5	6	4	15	3	0	1.00	0.83

TABLE III

EVALUATION FOR DATA-SET 3 [5176 FRAMES]

Action	II	III	IV	V	VI					
Number	6	6	6	5	7	TP	FN	FP	Precision	Recall
MCMC Method [11]	6	0	6	0	7	19	11	7	0.73	0.63
Proposed Method	5	0	2	0	2	9	21	4	0.69	0.3

TABLE IV

EVALUATION FOR DATA-SET 4 [16386 FRAMES]

Action	II	III	IV	V	VI					
Number	16	24	24	12	20	TP	FN	FP	Precision	Recall
MCMC Method [11]	2	11	11	0	19	43	53	38	0.53	0.45
Proposed Method	11	12	13	11	20	67	32	10	0.87	0.68

and global Dynamic Programming (DP) in polynomial time $O(N \log N)$, where N is a total amount of data.

The proposed method is not applicable to real-time applications because Kd-tree must be built on the entire observation. However, it still offers a powerful tool to off-line applications since a non-Markovian combinatorial optimization problem is solved in polynomial time due to the efficient computation algorithm.

The notion of meaningfulness is highly context-dependent and it is not practical to tackle it by the system alone. Future work should be to bring a user into the action understanding framework in interactive way, so that the system can naturally incorporate the knowledge of the users.

VI. ACKNOWLEDGMENTS

This study was supported by Program for Improvement of Research Environment for Young Researchers from Special Coordination Funds for Promoting Science and Technology (SCF) commissioned by the Ministry of Education, Culture, Sports, Science and Technology (MEXT) of Japan.

REFERENCES

[1] K. Ikeuchi and T. Suehiro, "Toward an assembly plan from observation part i: Task recognition with polyhedral objects," *IEEE Trans. Robotics and Automation*, vol. 10, no. 3, pp. 368–384, 1994.

[2] Y. Kuniyoshi, M. Inaba, and H. Inoue, "Learning by watching," *IEEE Trans. Robotics and Automation*, vol. 10, no. 6, pp. 799–822, 1994.

[3] J. Takamatsu, T. Morita, K. Ogawara, H. Kimura, and K. Ikeuchi, "Representation for knot-tying tasks," *IEEE Transactions on Robotics*, vol. 22, no. 1, pp. 65–78, 2006.

[4] S. Nakaoka, A. Nakazawa, K. Yokoi, and K. Ikeuchi, "Leg motion primitives for a dancing humanoid robot," in *Int. conf. on Robotics and Automation*, 2004, pp. 610–615.

[5] T. Inamura, Y. Nakamura, and I. Toshima, "Embodied symbol emergence based on mimesis theory," *Int. Journal of Robotics Research*, vol. 23, no. 4, pp. 363–377, 2004.

[6] T. Mori, Y. Nejigane, M. Shimosaka, Y. Segawa, T. Harada, and T. Sato, "Online recognition and segmentation for time-series motion with hmm and conceptual relation of actions," in *Int. conf. on Intelligent Robots and Systems*, 2005, pp. 2569–2574.

[7] T. Zhao, T. Wang, and H. Shum, "Learning a highly structured motion model for 3d human tracking," in *Proc. of Asian Conference of Computer Vision*, 2002.

[8] A. Boick and A. Wilson, "A state-based approach to the representation and recognition of gesture," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 12, pp. 1325–1337, 1997.

[9] K. Ogawara, J. Takamatsu, H. Kimura, and K. Ikeuchi, "Extraction of essential interactinos through multiple observations of human demonstrations," *IEEE Transactions on Industrial Electronics*, vol. 50, no. 4, pp. 667–675, 2003.

[10] S. Uchida, A. Mori, R. Kurazume, R. ichiro Taniguchi, and T. Hasegawa, "Logical dp matching for detecting similar subsequence," in *Proc. of Asian Conference of Computer Vision*, 2007.

[11] K. Ogawara, "Learning meaningful interactions from observation by repetitious motion analysis," in *The Third Joint Workshop on Machine Perception and Robotics*, 2007, pp. OS2–3.