2008 IEEE International Conference on
Robotics and Automation
Pasadena, CA, USA, May 19-23, 2008

Human-Inspired Robot Task Learning from Human Teaching

Xianghai Wu, *Student Member, IEEE,* and Jonathan Kofman, *Member*, *IEEE*

Abstract—**The ability of a service or personal robot to learn new tasks from human teaching is important if it is to be multi-functioning and serve users a lifetime. Considering the vast variation of tasks, work environments, and nature of potential teachers or users who may not have knowledge in robotics, the problem of task teaching and learning can be difficult to achieve. Current methods of robot teaching and learning do not yet enable the robot to learn different types of tasks from the teaching by a general user. This paper presents a human-inspired method of robot task learning from human instructive hand-to-hand teaching. The method is novel in including an introduction of the complete task to the robot before task demonstration, a voting algorithm for segmenting the demonstrated task trajectory, and a Bayesian approach to assign partitioned trajectory segments to subtasks. Also, the proposed trajectory blending scheme can generate actual task paths in real-time to adapt learned tasks to new task setups.**

I. Introduction

The ability of a service robot to continuously learn new tasks and environments from the teaching of a general user who may be unknowledgeable in robotics is necessary if the robot is to be multi-functioning and serve a lifetime. However, current methods of robot learning from human teaching do not enable the robot to learn different types of tasks from a general user. Methods of teaching tasks to a robot by a human include teaching by guidance (TbG), teaching by human demonstration (TbD), and human-style teaching (HST) [1]. A main limitation of all of these approaches is insufficient exploitation of the teacher's expertise in the tasks.

TbG methods involve having an operator (or teacher) directly move the robot arm or end-effector along a desired trajectory via a teach pendant, 6 degree-of freedom (DOF) mouse, joystick, force/moment sensors [2][3], hand gestures, and graphical-based, virtual reality [4]-[7] and kinesthetic [8] techniques. Although TbG methods are simple and effective, they are only suitable in well-structured work environments. Teaching by human demonstration (TbD), also called programming by demonstration (PbD), allows a teacher to demonstrate a task to a robot naturally and effectively [9]-[11]. TbD can also be performed through teleoperation by natural arm motion tracking [12][13] (called hand-to-hand teaching in this paper), using a vision-based interface [14][15] or tele-suit [12][13]. Recently, HST methods have been proposed, such as tutelage-like teaching [16], demonstration with instructions and timely feedback [17], and teaching by step-by-step instruction with rich mutual interaction [16][18]. These promising methods integrate natural and intuitive human-robot interaction with TbD techniques. However, these methods still fall short of effective and practical schemes to sufficiently exploit the task expertise of the teacher. Ideally, the teacher's expertise in task partitioning, understanding, and abstraction should be well exploited in natural and effective ways.

The simplest way to learn a task from human teaching is to repeat the taught task trajectories or determine the task paths by interpolating through demonstrated waypoints. Hidden Markov Model, Gaussian Mixture Model, fuzzy logic, and neural network techniques have been also used to approximate and generalize taught task trajectories [7][8] [19]-[22]. These machine learning techniques generally treat each trajectory as a whole piece, and need multiple trials or demonstrations. It is also very difficult for the robot to interpret and reuse the learned knowledge [18]. Generalization of task trajectories to permit a robot to adapt to different task setups has been somewhat achieved by first partitioning the taught task trajectories into different segments (or episodes) using motion breakpoints [23], and then generalizing each resulting episode [12][13]. [9] and [18] have further generated the task structure from the episodes. However, their methods of building task structures are not adequate for general tasks in the service domain. Robot adaptation to task setups different from the demonstrated setup has been achieved; however, it has been limited.

This paper presents a new human-inspired method of robot task learning from user teaching to provide a robot with the ability to adapt learned task information to new task setups. A step of overall task introduction (a brief description of the task structure), is carried out before demonstration of the task to exploit the teacher's expertise in the task. This gives the robot an overall top-down comprehension of the task to be taught and helps the robot construct a task structure that would be easily understood by new users. In addition, the teacher provides vocal cues of subtask partitions at subtask transitions during hand-to-hand task demonstration by teleoperation. New approaches are proposed to segment the taught task and assign partitioned task trajectory segments to the introduced subtasks. A method is also presented to adapt learned task trajectories to new and different task setups. The teacher is assumed to have expertise in the tasks to be taught and is required to pre-analyze the tasks carefully at the task level.

Manuscript received September. 14, 2007. This work was supported by the Natural Sciences, Engineering Research Council of Canada (NSERC) (grants and scholarship),and an Ontario Graduate Scholarship.

Xianghai Wu and Jonathan Kofman are with the Department of Systems Design Engineering, University of Waterloo, Waterloo, ON, N2L 3G1 Canada (e-mail: {x9wu, jkofman}@engmail.uwaterloo.ca).

This paper is organized as follows. The task introduction and instructive hand-to-hand robot-task teaching are presented in Section II. Methods of robot learning of taught tasks are proposed in Section III. In Section IV, experiments and results are presented and Section V concludes the paper.

II. Task Introduction and Instructive Hand-to-Hand Robot-Task Teaching

A. Overall Introduction of Task to be Taught

The introduction of the task to be taught is a step that is inspired from human teaching, where for example, an outline of a lecture would first be given before the details are taught. Before demonstrating a real task, the teacher would deliver a general introduction of the task structure or plan that could include the task name, the overall task goal, the number of its subtasks, and information of the objects and actions involved, using a graphical user interface (GUI) or voice dialogue. The user would further give a concise description about each subtask, and sub-subtask, which includes similar types of information as for the overall task. The robot would establish a sketch of the hierarchical task structure. The information of the involved objects and actions would serve as important cues for task partitioning (discussed in later sections). An example of a hierarchical task structure for a pick-and-place task is depicted in Fig. 1. This task will be used to explain the proposed methods in the following sections. After the task introduction, the teacher will demonstrate the task to the robot.

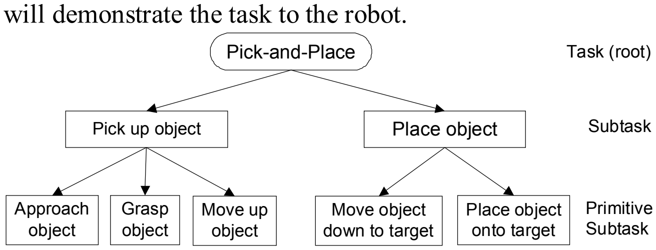

Fig. 1: Hierarchical representation of a pick-and-place task. The overall task (root node) consists of two subtasks (sub-nodes): “pick up object” and “place object”, which both consist of primitive subtasks.

B. Instructive Hand-To-Hand Robot-Task Demonstration

Here, the hand-to-hand teaching is achieved via teleoperation using natural arm motion tracking techniques. The main advantages of this approach over direct task demonstration by the teacher are that the goal oriented motion correspondences between the robot and teacher are solved by teleoperation and the demonstration is observed and understood from the robot’s own perspective [24]. The teacher is also permitted to teach the robot from remote sites, and this can be an advantageous for different applications such as telehealth and hazardous environments.

During task demonstration, the teacher mainly focuses on the demonstration so that the task can be demonstrated naturally and fluently while the teacher gives some simple utterances, such as “first step”, “then”, and “next step”, at the approximate subtask transitions based on their knowledge of the task. These simple organizational markers serve as useful cues for partitioning the task. The teacher also calls attention of the robot to changes of key states related to the task. At the same time, the robot actively observes the task, listens to the teacher’s instructions, quickly responds to the teacher’s requests, learns task-specific information such as alignments of objects, and records all pertinent data. After the demonstration, the robot has to learn the task, including the task trajectory and detailed task structure. The human teaching continues to be coupled with the robot learning process and task practice, until the robot has learned the taught task as expected.

III. Robot Learning of Taught Tasks

A. Vote-based Segmentation of Demonstrated Task Trajectory

Segmentation of a task trajectory has commonly been achieved using motion breakpoints [23][25], such as evident features of tactile feedback [27], instants of grasping and releasing [26], and mean-squared velocity (MSV) of the overall speed profiles [12][13]. Segmentation based on information from multiple sources may be more successful. In this paper, we present a vote-based algorithm to segment the taught task trajectory, which can easily accommodate votes from any signals.

Suppose m different signals are selected to segment the demonstrated task trajectory, for instance, the robot gripper status, mean squared velocity (MSV) of the robot end-effector, and the vocal task-partition cues given by the teacher during the task demonstration. Let T_i^k, $i = 1,\ldots,n^k$ and $k = 1,\ldots,m$, be the set of candidate segmentation time instants (e.g. instants that gripper status changed, or the MSV is at local minimum and less than its mean value) generated by analyzing the k^{th} signal, where n^k is the total number of partition instants from the given signal. A vote for segmenting the task motion by the k^{th} signal is given by

$$v_k(t) = \exp\left(-c_k\left(X(t)-X(T_i^k)\right)^2\right), t \in \left[\frac{T_{i-1}^k + T_i^k}{2}, \frac{T_i^k + T_{i+1}^k}{2}\right) \quad (1)$$

where $X(t)$ represents the recorded pose (position and orientation) of the end-effector at the t^{th} time sample; and c_k is a parameter that determines if the task trajectory should be partitioned in the vicinity of T_i^k based on analysis of the k^{th} signal. A greater c_k will influence the partition to be closer to T_i^k. The overall vote for segmenting the task motion is the weighted sum of the votes from all corresponding signals, and is defined by:

$$v(t) = \sum_{k=1}^{k=m} w_k v_k(t) \quad (2)$$

where w_k is the weight of the vote by the k^{th} signal, and is determined based on its relative segmentation reliability.

The overall segmentation candidates, can be determined as T_i, $i = 1,\ldots,n$, when $v(T_i)$ are local maxima greater than a threshold set to mean $v(t)$, where n is the total number of generated partition candidates.

False segmentation may occur at some instants. Let E_i be the i^{th} episode, or trajectory segment, $t \in [T_{i-1}, T_i]$. A false episode might be eliminated based on its probability that it is a valid segment, which is defined as a function of the following factors: whether the spatial and temporal length of the episode are long enough; whether the episode has clear characteristics, for example, primarily moving left/right /backward/forward/up/down, mainly turning left/right/up/ down, or moving very fast.

B. Assignment of Obtained Trajectory Episodes to Introduced Task Structure

The assignment of an episode to a particular primitive subtask depends on the joint probability that one should belong to or contain the other. Based on Bayes' rule, the joint probability can expressed as:

$$P(E_i, S_j) = P(S_j \mid E_i)P(E_i) \tag{3}$$

where S_j is the j^{th} primitive subtask, $i = 1,\ldots,n^E$; $j = 1,\ldots, n^S$, and n^E and n^S are the total numbers of episodes and primitive subtasks, respectively. $P(E_i)$ is the probability that E_i is a valid segment while $P(S_j \mid E_i)$ is the conditional probability that E_i should be assigned to S_j given E_i. $P(S_j \mid E_i)$ is a function of the following factors (used with predefined scales): (a) how well the characteristics of E_i match the action of the primitive subtask S_j, for example, if S_j is a subtask: "move object down to target", then the system can check whether the motion of E_i is mainly downward and the distance between the robot and target is considerably reduced; (b) whether salient events that occurred in this episode match the action of S_j; (c) whether the objects involved in this episode match the objects introduced in the primitive subtask S_j. E_i will be assigned to primitive subtask S_k with the following condition:

$$P(E_i, S_k) > P(E_i, S_j), \quad j = 1,\ldots, n^s, \forall\ j \neq k \tag{4}$$

Each primitive subtask should be assigned at least one episode. Otherwise, the learning system can either lower the threshold below mean $v(t)$ to obtain more segmentation candidates and repeat the above procedures, or replay the demonstrated task, state the obtained segmentation and assignment results to the teacher, and ask for teacher feedback.

C. Generalization of Demonstrated Task

A task motion is task-type dependent. The pick-and-place task (Fig. 1) is used as an example here to illustrate the proposed task generalization method; however, the method can be expanded to other types of tasks.

a) Determination of Docking Poses

A docking pose is defined relative to the object or target, and should be close to its corresponding grasping or releasing pose. Transformation from docking to grasping pose only allows translations along the approach direction and perpendicular to the approach and lateral directions, in the end-effector frame, whereas only the translation along the normal to the target surface is permitted from the docking pose to the final releasing pose. The two docking poses are the completion poses of subtasks "approach object" and "move object down to target", respectively.

b) Generalization of Task Trajectory in Different Frames

After having determined the docking pose for grasping and placing (or releasing), trajectories of subtasks "grasp object" and "place object onto target" have to be interpolated and re-sampled while trajectories of "approach object" and "move down to target" must also be altered to make related transitions smooth as in the demonstration. Then the new task trajectories will be further smoothed and re-sampled based on the motion information given in the demonstration and robot speed and acceleration constraints. This results in the generalized task trajectory in the world frame ($\mathbf{F}_W$), object frame ($\mathbf{F}_O$), and target frame ($\mathbf{F}_T$) respectively as:

$$\begin{aligned} \mathbf{\Gamma}_W &= \{X_i\}, \quad i = 1,..,h \\ \mathbf{\Gamma}_O &= \{{}^O X_i\}, \ i = 1,..,h \\ \mathbf{\Gamma}_T &= \{{}^T X_i\}, \ i = 1,..,h \end{aligned} \tag{5}$$

where X_i is the i^{th} robot end-effector location of the generalized task trajectory expressed in $\mathbf{F}_W$. ${}^O X_i$ and ${}^T X_i$ are the X_i in $\mathbf{F}_O$ and $\mathbf{F}_T$ respectively. h is the total number of sampled trajectory points. Similar methods were developed in [13], but multiple task demonstrations were required because two radial basis function networks were used to generalize taught trajectories in the world and object frames, respectively. Only one demonstration is used in this paper.

c) Generalization of Task Structure

Generalization of the task structure includes determination of the pre-condition, until-condition (subtask goal) and associated actions or operations for each subtask, and execution orders of these subtasks (e.g. required start and end states of the robot hand and objects of interest, alignment between objects and targets, spatial relationships between the robot and objects, and involved robot hand actions (close and open)). The relationships between the subtasks (dependencies of related pre-conditions and until-conditions) affect their execution order. For sequential subtasks, their execution order cannot be altered, while for parallel subtasks their order is flexible. Some subtasks may be optional, for example, when there is no key state change. The robot may ask the teacher to confirm or clarify the relationships between the subtasks.

D. Practice and Execution of Learned Tasks

a) Adaptation of Learned Task Trajectories to Actual Task Setups

Lieberman [13] proposed a promising solution to adapt a learned path to changing environments, using a blending mechanism. However, their method did not consider rotations or different possible robot starting poses. The principle of blending is that the closer the robot is to the object/target, the more closely the taught trajectory generalized in the object/target frame has to be followed. The actual task path in this paper is generated as follows:

$$\widetilde{X}_i = (1-\lambda_1(i))(\lambda_2(i)Y(t) + (1-\lambda_2(i))X_i) + \lambda_1(i)\,{}^{O}X_i,\ i \le h_O \quad (6)$$
$$\widetilde{X}_i = (1-\lambda_1(i))(\lambda_2(i)\,{}^{O}X_i + (1-\lambda_2(i))X_i) + \lambda_1(i)\,{}^{T}X_i,\ i > h_O$$

where $Y(t)$ is the current location of the robot, and h_0 is the time sample index at which the object is grasped. The object/target frame is updated based on real task setup. After the object is grasped, $Y(t)$ is substituted by ${}^{O}X_i$ because it is desirable to follow a similar departure path (relative to the object location before being grasped) as that in the demonstration. $\lambda_l(i)$ and $\lambda_2(i)$ are blending functions, and depend on the nature of the subtask that contains the path point X_i. For example, $\lambda_l(i)$=1 at the end of the subtask "approach object" because the robot must be at the docking pose at the end of the subtask. $\lambda_2(i)$ = 1 at i = 0 and rapidly decreases to 0 while $\lambda_2(h_0)$ = 1, and quickly drops to 0 afterward. The influence of $Y(t)$ immediately after the start of the task and of Γ_O immediately after the object is grasped is thus limited to a short period. A distance-based six-order polynomial function, adapted from [28], is used to calculate $\lambda_l(i)$ and $\lambda_2(i)$.

b) Practice of Learned Task and Refinement of Task Knowledge Based on Timely Feedback

The robot may practice the learned task at two different speeds, the first time, at a slower speed. Before performing each subtask, the robot tells the teacher the subtask name, precondition, until-condition, involved actions and objects, and expected key state changes. When practicing the subtask, the robot also points out its perceived key state changes. Then, the robot practices the task at normal speed for different task setups, and does not disclose its internal states intentionally. The teacher gives timely feedback or comments using utterances such as "move faster/slower" for speeds, "move more left/right/up/down/ forward/backward" for translation offsets, "turn more left/right/up/down/ forward/backward" for orientation offsets, "open/close hand", "start/pause/stop", or even voice feedback on the task structure. The robot can then refine its task knowledge according to the feedback received by changing, for example, its speed or step increment. The practice-feedback-refine cycle has to be repeated until both the robot and teacher agree that either the process of task teaching and learning has been accomplished or they must go back to some previous teaching or learning stages, modify related knowledge, and practice the task again.

IV. Experiments and Results

A. Experimental Setup

The robot used in the experiments was a 6-DOF Thermo CRS A465 manipulator (Fig. 2). A vision-based human-robot teleoperation system [14] was employed to hand-to-hand teach the robot the task along with speech recognition and text-to-speech engines (MS Speech SDK 5.1).

The task to be taught was a pick-and-place task, where a block was picked up and then placed on a target so that the corners and edges of the object would align with those of the target, respectively, as shown in Fig. 2. Currently, the robot does not have the ability to estimate the locations of the object and target. This information was provided to the robot manually before the robot began the practice and execution of the taught task. The robot recorded the demonstrated trajectory, robot gripper status, and teacher's vocal task-segmentation cues with a sampling period of 0.5 s.

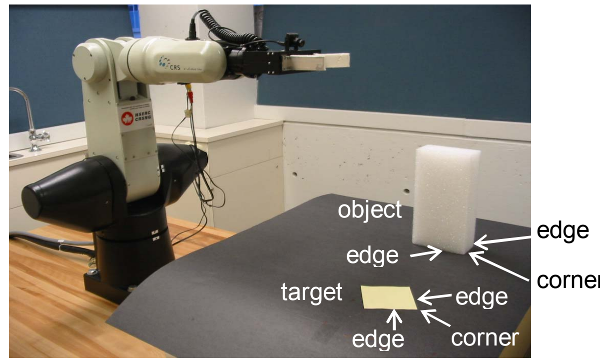

Fig. 2. Six-axis robot manipulator and task demonstration setup.

B. Experimental Results

The task structure, shown in Fig. 1, was introduced to the robot via voice dialogue under robot control. For example, after being told that the task had two subtasks, the robot asked for information about each subtask and each sub-subtask to gradually build the task structure tree.

MSV of the taught end-effector trajectory, gripper status, and vocal task-partition cues were used to segment the taught task trajectory. The overall vote $v(t)$, (Fig. 3), was computed using (1) and (2) with respective parameters c_{MSV} =1/(50$\underline{V}^2$), $c_{Gripper}$ = 1/(2$\underline{V}^2$), c_{Voice} = 1/(800$\underline{V}^2$), w_{MSV} = 0.19, $w_{Gripper}$ = 0.52, and w_{Voice} = 0.29, where $\underline{V}$ = 3.68 mm/s is the average magnitude of the velocity of the taught trajectory. Votes by the selected local minima of the MSV, teacher's vocal partition cues, and gripper status changes are shown in the figure. The survived and eliminated task trajectory segmentation points (SPs) are also illustrated in Fig. 3.

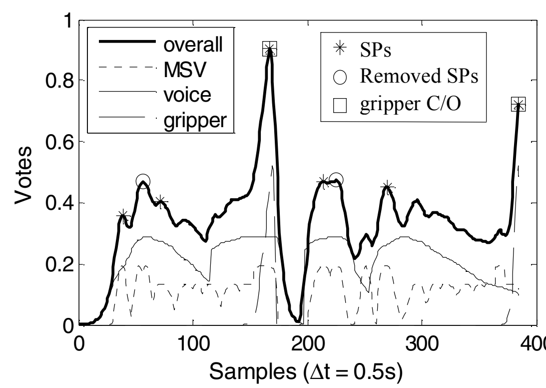

Fig. 3. Votes for segmenting the demonstrated task trajectory, showing overall vote, and votes by: MSV, voice cues, and gripper state changes. Resulting segmentation points (SPs) are indicated by the asterisk *, removed SPs by ○, and Gripper Close/Open action by □.

The generalized task trajectory in the world frame, Γ_W, and its taught counterpart are shown in Fig. 4. The resulting episode-subtask assignments are also depicted. Subtask "approach object" consists of two internal episodes delimited by marker ◆. The computed blending weights $\lambda_1(i)$ and $\lambda_2(i)$ are shown in Fig. 5. Before finishing the primitive subtasks "approach object" and "move object down to target", $\lambda_1(i)$=1 had to be true because the robot must reach the relevant docking poses. $\lambda_2(i)$ decreased rapidly to zero before $\lambda_1(i)$ increased to 0.5.

Generalization of the task structure was performed without difficulty. The primitive subtask "move up object" could be skipped as no associated key state changes (e.g. gripper or object status) occurred in this subtask.

The learned task was first practiced with half of the demonstrated velocity and the task setup (locations of the object, target, and robot starting point) similar to that in the demonstration. The practice performance did not require any negative feedback to be issued by the teacher. The robot then practiced the task at the full speed of the demonstration, using four different task setups. The robot succeeded in all setups. The tested task setups and final alignment errors between the object and target are given in Table 1. The final alignment results were rather excellent, as was expected due to the blending scheme.

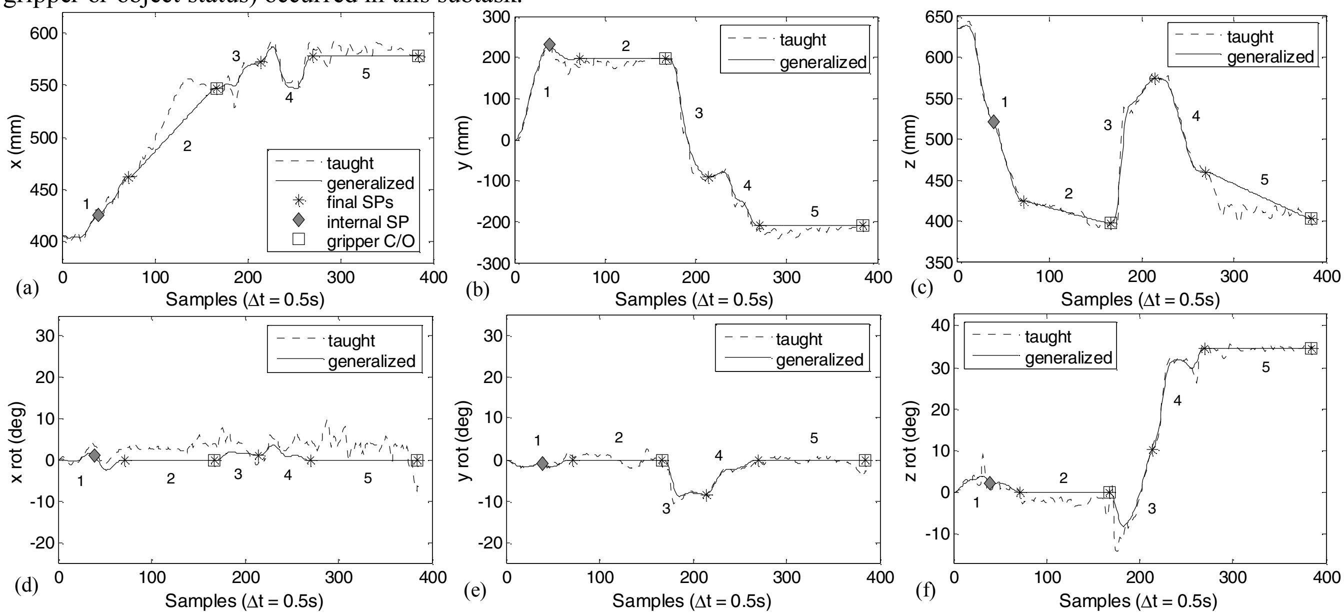

Fig. 4. Taught trajectories and their generalized counterparts in the world frame. Episodes 1,2,3,4 and 5, which are delimitated by asterisk * markers, are assigned to primitive subtasks: "approach object", "grasp object", "move up object", "move object down to target" and "place object on target", respectively. "Approach object" consists of two internal episodes delimited by marker "◆" in the figure.

TABLE 1. DIFFERENT PRACTICED TASK SETUPS AND CORRESPONDING RESULTS. The alignment errors consist of the distance (Δ_d) between the two corners, and orientation errors (Δ_γ) between the edges. The setup in the demonstration was (x, y, z, α, β, γ): object at (560.3, 196.8, 396.7, 0, 0, 0), target at (588.8, -201.7, 403.0, 0, 0, 34.7), and robot starting point (406.4, 0, 635.0, 0, 0, 0)

Task Setups ($x, y, z, \alpha, \beta, \gamma$) (mm, mm, mm, deg, deg, deg)			Alignment errors (Δ_d, Δ_γ) (mm, deg)
Object Loc	Target Loc	Robot start Loc	
533.0,270.0,300 0.0, 0.0, 0.0	427.6,-230.3,300 0.0, 0.0, -15.0	406.4, 0.0, 635.0 0.0, 0.0, 0.0	0.0, 0.1
553.0,190.0,300 0.0, 0.0, 0.0	520.0,-248.6,300 0.0, 0.0, 40.0	406.4, 0.0, 635.0 0.0, 0.0, 0.0	0.5, 0.0
547.6,275.5,300 0.0, 0.0, -35.0	554.5,-187.4,300 0.0, 0.0, 55.0	406.4, 0.0, 635.0 0.0, 0.0, 0.0	1.0, 0.0
533.0,270.0,300 0.0, 0.0, 0.0	523.0,-239.9,300 0.0, 0.0, 5.0	510.0,-241.0,300 0.0, 0.0, 5.0	1.0, 0,0

C. Discussion of Experimental Results

The experimental results show that the proposed human-inspired robot task learning from the human instructive hand-to-hand teaching approach is practical and promising. The overall task introduction not only aids the robot to appreciate the task to be learned in a top-down way and build a task structure that other users can later understand easily. It also provides important cues to partition the task.

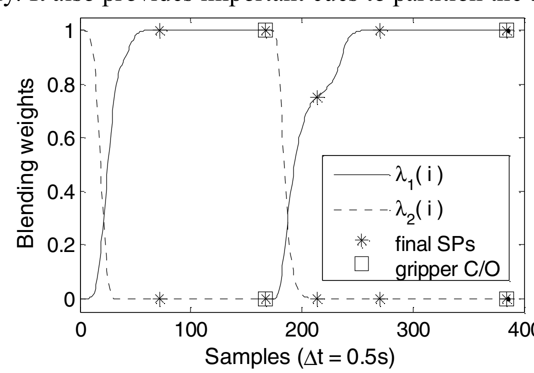

Fig. 5. Blending weights of the trajectories generalized in the object/target frame, $\lambda_1(i)$; and of the current robot end-effector location or departure trajectory after the object was grasped, $\lambda_2(i)$.

The hand-to-hand task demonstration was effective and intuitive to the teacher, although, significant concentration was required to complete the teleoperation task. The proposed voting algorithm for segmenting the demonstrated task motion can utilize different types of signals. For each signal, only its relative reliability for segmentation (its voting weight) and its sensitivity to possible partition points (its c in (1)) have to be determined. The proposed vote-based task trajectory segmentation and Bayesian rules-based episode-subtask assignment were used successfully.

The robot task practice showed the capacity of the robot to adapt the learned task to different task setups. The robot successfully applied its learned task to different locations of the object, target and robot starting points, even though the new locations were significantly different from those of the demonstration. The fourth setup in Table 1 had a robot starting position 0.43 m from that during task demonstration.

V. CONCLUSIONS

A human-inspired method of robot task learning from human instructive hand-to-hand teaching was proposed to let a user teach tasks to a service robot intuitively and effectively. By listening to the overall task introduction, vocal subtask-segmentation cues, and timely feedback during task practice, and by watching a task demonstration, it is possible for the robot to learn a task (including the task structure and trajectory) from human teaching, and organize the learned task knowledge in ways that facilitate interaction with general users. The main contributions include a step of overall task introduction before the task demonstration, a voting algorithm to partition the task trajectory, and a Bayesian rule-based method to assign task trajectory episodes to primitive subtasks. Future work includes increasing the sensing ability of the robot so that it can perceive its working environment, incorporating blending schemes while considering mechanical constraints of the robot, and teaching the robot more complex tasks by different subjects.

REFERENCES

[1] G. Biggs and B. MacDonald, “A survey of robot programming systems”, *Proc. of the Australasian Conf. on Robotics and Automation*, CSIRO, Brisbane, Australia, 2003.

[2] S. Muto and K. Shimokura K, “Teaching and control of robot contour-tracking using contact point detection”, in *Proc. 1994 IEEE Int. Conf. Robotics and Automation*, vol. 1, pt. 1, pp. 674-81, 1994.

[3] M. H. Choi and W. W. Lee, “Quantitative evaluation of an intuitive teaching method for industrial robot using a force/moment direction sensor”, *Int. J. of Control, Automation and Systems*, vol. 1, no. 3, pp. 395-400, Sep. 2003.

[4] Y. Kunii and H. Hashimoto, “Tele-teaching by human demonstration in virtual environment for robotic network system”, in *Proc. IEEE Int. Conf. on Robotics and Automation*, vol. 1, pt. 1, pp. 405-10, 1997

[5] R.M. Voyles, J.D. Morrow and P.K. Khosla, “Towards gesture-based programming: shape from motion primordial learning of sensorimotor primitives”, *Robotics and Autonomous Systems*, vol. 22, no. 3-4, pp. 361-75, Dec. 1997.

[6] M. Uechi, H. Ogata, Y. Nakamura and M. Mizukawa, “A component architecture for customizing robot-teaching systems”, in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, vol. 3, pt. 3, pp. 1625-32, 1999.

[7] J. Aleotti and S. Caselli, “Robust trajectory learning and approximation for robot programming by demonstration”, *Robotics and Autonomous Systems*, vol. 54, no. 5, 31 May 2006, pp. 409-13

[8] S. Calinon, F. Guenter, and A. Billard, “On Learning, Representing, and Generalizing a Task in a Humanoid Robot”, *IEEE Trans. Syst, Man, Cybernetics, Part B*, vol. 37, no. 2, pp. 286-98, Apr. 2007

[9] Y. Kuniyoshi, M. Inaba and H. Inoue, “Learning by watching: extracting reusable task knowledge from visual observation of human performance”, *IEEE Trans. on Robotics and Automation*, vol. 10, no. 6, pp. 799-822, Dec. 1994

[10] R. Dillmann, O. Rogalla, M. Ehrenmann, R. Zöllner, and M. Bordegoni, “Learning robot behaviour and skills based on human demonstration and advice: the machine learning paradigm”, in *Proc. 9th Int. Symp Robotics Res*, Snowbird, Utah, USA, 9-12. Oct 1999.

[11] R. Dillmann, “Teaching and learning of robot tasks via observation of human performance”, *J. of Robotics and Autonomous Systems*, vol. 47, no. 2-3, pp. 109-116, 2004.

[12] R.A. Peters, C.L. Campbell, W.J. Bluethmann and E. Huber, “Robonaut task learning through teleoperation*”, IEEE Int. Conf. on Robotics and Automation*, pt. 2, vol. 2, pp. 2806-11, 2003

[13] J. Lieberman, Teaching a Robot Manipulation Skills through Demonstration, M Sc thesis, Dept. of Mech. Eng., MIT, 2004.

[14] J. Kofman, X. Wu, T. Luu and S. Verma S., "Teleoperation of a robot manipulator using a vision-based human-robot interface", *IEEE Trans. on Industrial Electronics*, vol. 52, no. 5, pp. 1206-1219, 2005.

[15] J. Kofman, S. Verma, and X. Wu, “Robot-manipulator teleoperation by markerless vision-based hand-arm tracking”, *Int J Optomechatronics*, vol. 1, no 3, pp. 331-357, 2007.

[16] A. Lockerd and C. Breazeal, “Tutelage and socially guided robot learning”, *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pt. 4, vol. 4, pp. 3475-80, 2004.

[17] M. N. Nicolescu and M. J. Mataric, “Natural methods for robot task learning: instructive demonstrations, generalization and practice”, *Proc. of Int. Conf. on Autonomous Agents*, vol. 2, 2003, pp. 241-248.

[18] C. Breazeal, A. Brooks, J. Gray, G. Hoffman, C. Kidd, H. Lee, J. Lieberman, et al., "Tutelage and collaboration for humanoid robots", *Int J Human Robot*, 2004, vol. 1, no. 2, pp. 315–348

[19] F. Hoffmann, “Fuzzy behavior coordination for robot learning from demonstration”, *NAFIPS Annual Meeting of the North American Fuzzy Information Processing Society*, pt. 1, pp. 157-62 vol. 1, 2004

[20] P. Martin and U. Nehmzow, “Programming by teaching: neural network control in the Manchester mobile robot”, in *Proc 1995 2nd IFAC Conf. on Intelligent Autonomous Vehicles*, 1995, pp. 277-82.

[21] D.R. Myers, M.J. Pritchard and M.D.J. Brown, “Automated programming of an industrial robot through teach by showing”, in *Proc. 2001 ICRA. IEEE Int. Conf. on Robotics and Automation*, vol. 4, pt. 4, pp. 4078-83, 2001.

[22] S. Calinon and A. Billard, “Active Teaching in Robot Programming by Demonstration”, in Proc. Of the 16th IEEE Int. Symp on Robot and Human interactive Communication, pp. 702-7, Aug. 2007.

[23] S.K. Tso and K.P. Liu, “Automatic generation of robot program codes from perception of human demonstration”, *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots*, pt. 1, vol. 1, pp. 23-8, 1995.

[24] C. Breazeal, M. Berlin, A. Brooks, J. Gray, A.L. Thomaz, “Using perspective taking to learn from ambiguous demonstrations”, *Robot and Autonomous Systems*, vol. 54, no. 5, 31 May 2006, pp. 385-93

[25] M. Yeasin and S. Chaudhuri, “Automatic robot programming by visual demonstration of task execution”, *Proc. 8th Int. Conf. on Advanced Robotics ICAR'97*, pp. 913-18, 1997.

[26] N. Delson and H. West, “Robot programming by human demonstration: the use of human variation in identifying obstacle free trajectories”, in *Proc. 1994 IEEE Int. Conf. on Robotics and Automation*, vol. 1, pt. 1, pp. 564-71, 1994.

[27] M. Ehrenmann, R. Zöllner, O. Rogalla, S. Vacek and R. Dillmann, “Observation in programming by demonstration: training and execution environment”, *Int. Conf. on Humanoid Robots HUMANOIDS*, Karlsruhe, Munich, Germany, October 2003.

[28] R. Volpe, “Task space velocity blending for real-time trajectory generation”, in *Proc of IEEE Int Conf on Robotics and Automation*, vol. 2, pp. 680-687, 1993.