Using Human Motion Estimation for Human-Robot Cooperative Manipulation

Anand Thobbi, Ye Gu, Weihua Sheng, *Senior Member, IEEE*

***Abstract*— Traditionally the leader or follower role of the robot in a human-robot collaborative task has to be predetermined. However, humans performing collaborative tasks can switch between or share the leader-follower roles effortlessly even in the absence of audio-visual cues. This is because humans are capable of developing a mutual understanding while performing the collaborative task. This paper proposes a framework to endow robots with a similar capability. Behavior of the robot is controlled by two types of controllers such as reactive and proactive controllers. The reactive controller causes the robot to behave as a follower and the proactive controller causes it to behave as the leader. The proactive controller suggests proactive actions based on human motion prediction. The framework relies on a novel technique to compute a measure of confidence for the prediction. This confidence measure determines the leader/follower role of the robot. Hence, the robot can switch roles during the task autonomously and dynamically. A table-lifting task which is essentially a cooperative manipulation task is considered to demonstrate the proposed framework. Finally, the performance of the human-robot team carrying out this task is experimentally evaluated. Results show that the proposed system improves the overall task performance.**

I. INTRODUCTION

For service robots to be useful, one of the fundamental abilities they should possess, is to work collaboratively with humans. A common example where collaboration would be required, is in a cooperative manipulation task, where the human-robot team has to manipulate an object of interest. Technology that enables robots to work collaboratively with humans can be widely applied in the industry as well as in common day-to-day scenarios. This field has seen a renewed interest in recent years because of the possibility of humanoid robots residing along with us in the near future.

Traditionally, the intellectual responsibility of planning and guiding the co-operative task is placed entirely on the human while the collaborating robot is assigned a mere follower role. These robot followers are pre-programmed with simple reactive behaviors. For example, a popular approach for accomplishing a cooperative object manipulation task, is using impedance control [1], [2]. However adopting such a naive strategy requires the human to spend extra energy in dragging the robot, apart from the energy spent in moving the load itself. Furthermore, a goal such as keeping the table exactly horizontal throughout the table-lifting task is very difficult to achieve using this technique alone.

Anand Thobbi, Ye Gu and Weihua Sheng are with the Department of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK 74074, USA. [thobbi@okstate.edu, ye.gu@okstate.edu, weihua.sheng@okstate.edu]

Maeda *et al.* were amongst the earliest to provide a solution to this problem, by using a human motion prediction technique which enables the robot partner to work *proactively* with the human [3]. Human motion prediction was obtained by following the assumption, that the fellow human's motion satisfies the minimum jerk model [4] in the cooperative manipulation setting. Based on estimation of the minimum jerk model parameters, the robot could predict the velocity profile of the human's motion, which could then be used to take a proactive action. This strategy was shown to reduce the human's effort for the cooperative manipulation task. Recently, we have seen a resurgence in the studies of physical human robot interaction which make use of motion prediction strategies. Corteville *et al.* presented a robot assistant which could predict the humans motion using an extended Kalman filter (EKF) [5]. The EKF was designed according to the minimum jerk model. The amount of assistance provided by the robot throughout the entire task had to be decided beforehand. In [6] the authors proposed a solution to change the role of the robot during the task execution using a homotopy switching model, although manually. Automatic adjustment of the homotopy variable α_i which decides the role of the robot was left as an open question, for which the proposed work offers a solution. Another shortcoming in [5] and [6] is the assumption that the robot should know the destination of the object being transported so that a plan of motion could be generated. If the destination is changed mid-way, a new subtask has to be generated on the fly which is non-trivial and is a separate work in itself [7]. Apart from cooperative tasks, human motion prediction has also been applied extensively in robotic teleoperation tasks [8], [9], [10].

Recent works show that the minimum jerk model may not be suitable for cooperative manipulation tasks [11]. The minimum jerk model assumption fails when there are large perturbations in the motion trajectory, or if the human decides to change the course of the trajectory during the task execution. In such cases, the robot might fail to comply with the human, which may lead to disastrous consequences. Also, in order to apply the minimum jerk model successfully, the final position of the object must be known both to the human and robot which is cumbersome in real world situations. It is interesting to note that two humans can excel in a table-lifting task even if one does not know the final position of the object. Other related work include [12] which proposes a task-model learning approach combined with an adaptive control system. After going through a two-step learning process, the robot can work collaboratively with the human while inferring his

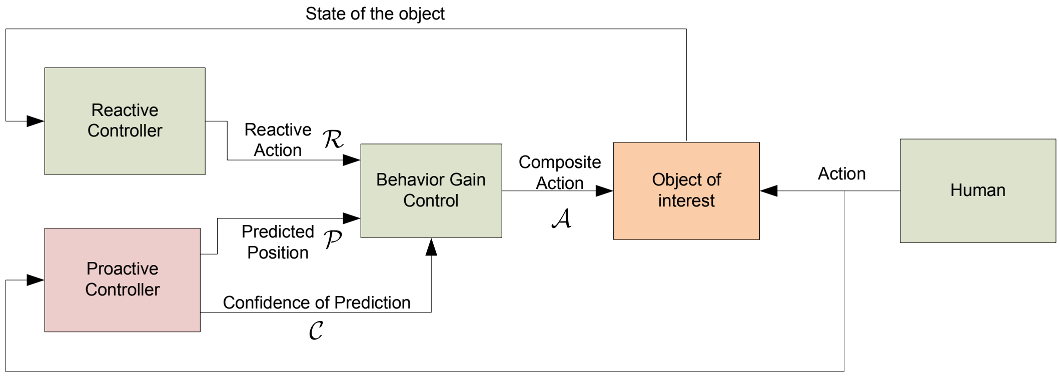

Fig. 1. Proposed approach block diagram

intent.

In this work, we propose a novel solution to address the problem of switching the robot's role automatically during a cooperative manipulation task. Additionally, the robot does not need to know the final position of the object. This is practically desirable, since the motion trajectory of the object may require to be changed during task execution, depending upon the environment, obstacles or the physical limitations of the human and/or the robot.

The proposed work uses a prediction-evaluation method to estimate the confidence of prediction and using it to adjust the role of the robot. Our hypothesis stems from the observation that, in a human-human team performing a collaborative task, each human constantly predicts the other's motion. Based on how well the other person conforms to his predictions, the human can decide whether to lead him or follow him. We apply the same strategy to the humanoid robot. Another way of looking at this solution is, suppose if the robot is able to predict the human's motion accurately, it means that the robot has acquired an accurate model of the human's behavior. Hence, it can start behaving as a leader and proactively take the next action based on its prediction. However if the robot has not been able to predict the humans motion correctly, it is better for the robot to reactively comply with the human. This intuition sets the basis for adjusting the leader/follower role of the robot continuously and dynamically.

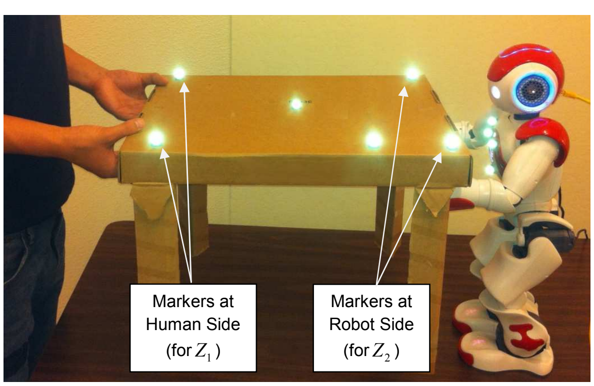

Fig. 2. Experimental Setup

The rest of the paper is organized as follows. In section II we present the experimental platform. Section III presents the methodology of the proposed system. The results are presented in section IV. Section V concludes the paper with a few discussions and future works.

II. Experimental Platform

For the experiments we developed a platform which consists mainly of a Vicon motion capture system and a Nao humanoid robot. The table-lifting task consists of the human and humanoid robot lifting up a dummy table to a random height and keeping it down. Fig. 2 shows the experimental setup. Only the positional information of the table is used for characterizing the task. We do not use force sensors because the table does not have a significant weight. The Vicon motion capture system provides precise position and motion information about the table. Motion of the robot hand is constrained to 1-D up-down motion. However, the proposed system can be easily extended to handle multiple dimensions. C++ is used at the front-end for communicating with the robot and MATLAB©is used at the back-end for processing data.

A. Motion Capture System

The Vicon MX motion capture system is used for obtaining the positional information of the table, needed for carrying out the task as well as for evaluating the performance. Frame rate of the motion capture system is 100 Hz. The robot can acquire motion capture data only after it completes the commanded action. Hence, the frame-rate is limited by the speed of the robot to complete a commanded action which is typically 100 ms. The motion capture system is guaranteed to be precise within 0.7 mm. Figure 2 also shows the table with the markers. The absolute position of the table along the vertical axis is calculated by averaging the positions of the markers placed at the end points. Let Z_1 and Z_2 be the instantaneous 1-D coordinates of the human-end and the robot-end of the table respectively. Observation and prediction is done on Z_1.

B. Humanoid Robot

We use the medium sized Nao humanoid robot manufactured by Aldebaran robotics. An inverse kinematic procedure

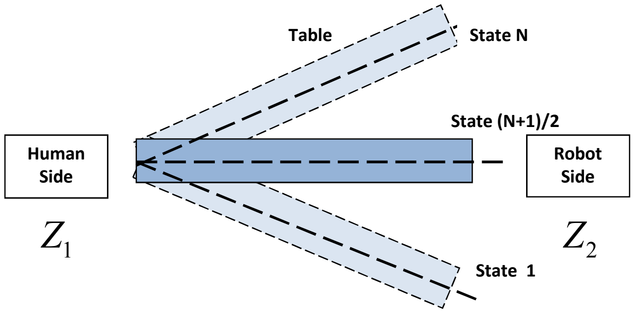

Fig. 3. State representation for reinforcement learning

provided by the SDK is used to control the end effector (hand-tip) position of the robot. Since the robot control is based only on controlling the end-effector position, certain offset had to be added in order to compensate for the small, but definite weight of the table.

III. METHODOLOGY

Figure 1 shows the conceptual block diagram for the proposed system. The framework consists of the reactive controller, proactive controller and the behavior gain control blocks. As the name suggests, the reactive controller generates a reactive robot behavior based upon the current state of the environment. The proactive controller consists of an EKF based human motion predictor and an evaluation-based confidence generator. Based upon the observed human actions, the predictor estimates the position of the human in the next time-step, which decides the robot's proactive action. Additionally, it generates the confidence of prediction, which is the key in adjusting the role of the robot. Based upon the confidence value, the behavior gain control blocks mixes the reactive and proactive actions to generate a composite action which is taken by the robot. According to our hypothesis, the weight allotted by the gain control block to the proactive behavior varies directly as the confidence value. In the remainder of this section, we discuss the details of the proposed framework.

A. Reactive Controller

The reactive controller generates a reactive response by the robot to the observed state of the object. In the table-lifting task, this controller observes the position of the table and suggests a suitable action to perform so that a certain objective is achieved. For our experiments, the objective is to keep the table horizontal throughout the task. This can be accomplished using any generic feedback controller. However, we chose to use a controller learned from reinforcement learning for the following reasons :

- It is possible to learn a good controller in a short time.
- It compensates for the time needed to manually tune the parameters of a feedback controller.
- Objective of the task is very simple in the current experiment. However, in the future, we will consider complex tasks such as *keeping a bowl in the center of the table* while performing the table lifting task. Complex tasks like these, have a long term reward to maintain for which reinforcement learning is most suited. Also, such high level objectives are much easier to specify using reinforcement learning.

In this work we use the discrete Q-learning algorithm. The Q-table update equation is given by

$$\Delta Q(s_t, a_t) = \alpha[r + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t)] \quad (1)$$

where r is the reward, α is the learning rate and γ is the discount factor. For the task at hand, γ does not play a significant role, since there is no sense of a long term reward. The state of the environment is determined by the incline of the table at the given moment. This information is obtained from the motion capture system. Incline of the table is quantized into discrete number of states. Figure 3 shows a state space consisting of N states. The action space consists of a predetermined discrete set of commands which move the robot's hand-tip up or down by specified distances. The robot has to undergo an online learning phase to learn the Q-table. During this phase, it is assumed that the human remains comfortably stationary. To speed up the learning phase we use a simple guided reinforcement learning algorithm based on counting the number of state-action visits. Essentially, the action selection for exploration is done on the basis of number of visits to the particular state-action pair, instead of random action selection as in ϵ - greedy algorithms. The reinforcement learning algorithm is given below.

Algorithm 1 Guided Reinforcement Learning

Initialize $Visit(s_i, a_i) = 0 \ \forall i \in N$
Initialize Q-table $Q(s_i, a_i) = 0 \ \forall i \in N$
while Learning phase **do**
    $t = timestep$
    $s_t = getState()$
    Select $a_t \leftarrow$ argmin$(a)[Visit(s_t, a)]$
    Take action a_t
    $Visit(s_t, a_t) \leftarrow Visit(s_t, a_t) + 1$
    $r = getReward()$
    Update $Q(s_t, a_t)$ using (1)
end while

B. Proactive Controller

The proactive controller is the most important block of the proposed system. Role of the proactive controller is to keep a track of actions performed by the human and generate a prediction of the human's position in the next time-step, along-with a confidence measure for the prediction. For the prediction purpose, an extended Kalman filter is used. State of the EKF x_k is given by

$$x_k = \begin{pmatrix} s_k \\ v_k \\ a_k \end{pmatrix} \quad (2)$$

The measurement model is given by

$$z_k = s_k + v \tag{3}$$

where s_k is the displacement of the human's end of the table (equivalently his hand-tip), v_k is his velocity, a_k is his acceleration and $v \sim N(0, R)$ is the measurement noise, all at the instant k.

We use the assumption that the acceleration of the human hand changes slowly throughout the motion since humans naturally try to minimize jerk. Note that this is not the same as using the minimum jerk model.

Hence, the state update equation can be written as

$$x_{k+1} = \begin{pmatrix} s_k + v_k t + \frac{1}{2} a_k t^2 \\ v_k + a_k t \\ a_k \end{pmatrix} + w \tag{4}$$

where $w \sim N(0, Q)$ is the process noise.

Based on the state estimate $\hat{x}_k$, the human's position at the next time-step can be predicted as

$$\hat{s}_{k+1} = \hat{s}_k + \hat{v}_k t + \frac{1}{2} \hat{a}_k t^2 \tag{5}$$

The variance of the measurement noise (R) is initialized to 0.7 which corresponds to the uncertainty in measurement obtained by the Vicon system. Using this EKF, it is possible to get nearly accurate predictions of the human's motion.

For obtaining the confidence of prediction, we derive inspiration from [13], wherein the authors proposed a technique, to obtain a confidence measure based on the statistical properties of the residuals between the predicted measurements and the observed measurements. In our technique, the EKF provides a state estimate and an associated covariance matrix. Firstly, we marginalize the covariance matrix to include only the 1-D variance associated with the position prediction, say ρ. Let the predicted position be $\hat{s}_k$. Then, we evaluate the likelihood of the observed measurement, z_k using an unnormalized Gaussian distribution given by

$$\mathcal{L}_k = \exp\left(-\frac{(z_k - \hat{s}_k)^2}{2\rho^2}\right) \tag{6}$$

We choose an unnormalized Gaussian distribution to make $0 < \mathcal{L} \leq 1$. It can be seen that $\mathcal{L}$ would give us a direct measure of confidence about the prediction based on the evaluation of the previous prediction against the true measurement. However, considering only the last step measurement error is not sufficient. For the confidence measure, we introduce a function given by

$$\mathcal{C}_{k+1} = \frac{\mathcal{L}_k + \phi\mathcal{L}_{k-1} + \cdots + \phi^{k-1}\mathcal{L}_1}{1 + \phi + \cdots + \phi^{k-1}} \tag{7}$$

The subscripts denote the time-steps at which they were obtained. Hence, $\mathcal{C}_{k+1}$ is the confidence of prediction for the next time-step, that considers all the likelihoods observed previously, weighted by the forgetting factor ϕ, where $0 < \phi \leq 1$. This function can be implemented recursively. Also, it can be seen that the denominator is for normalization.

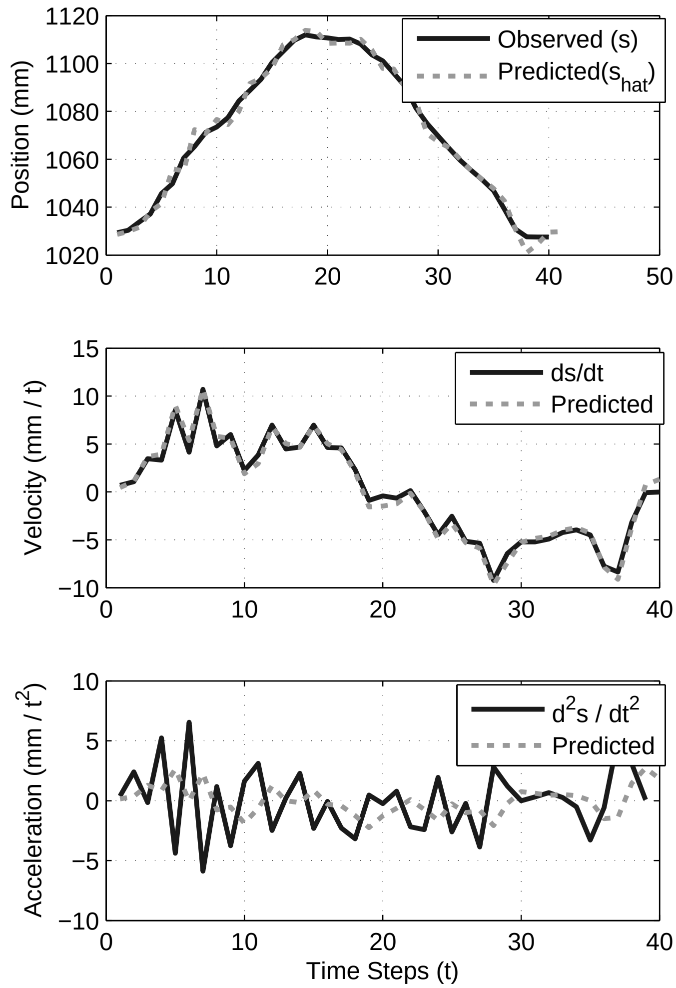

Fig. 4. Predictions obtained from EKF

C. Behavior Gain Control

At a given time step k, let the reactive controller suggest a next-step action $\mathcal{R}_{k+1}$ and the proactive controller suggest an action $\mathcal{P}_{k+1}$. Let the confidence of this prediction be $\mathcal{C}_{k+1}$. The gain control block combines these together to form a composite action $\mathcal{A}_{k+1}$ given by

$$\mathcal{A}_{k+1} = \mathcal{C}_{k+1}\mathcal{P}_{k+1} + (1 - \mathcal{C}_{k+1})\mathcal{R}_{k+1} \tag{8}$$

This action is taken by the robot at time-step $k + 1$. The inspiration for this form has been taken from [6]. Note that because $0 < \mathcal{C} \leq 1$, the robot does not act as a pure leader or pure follower, but has characteristics of both in different amounts.

If the confidence of prediction $\mathcal{C}_{k+1}$ is high, larger weight is allotted to the proactive action. Hence, the robot's action has leader-like characteristics. If the robot is not very confident about the prediction, larger weight is allotted to the reactive behavior and the robot's action seems follower-like. Since the system works in real time, the change of behavior is dynamic and automatic.

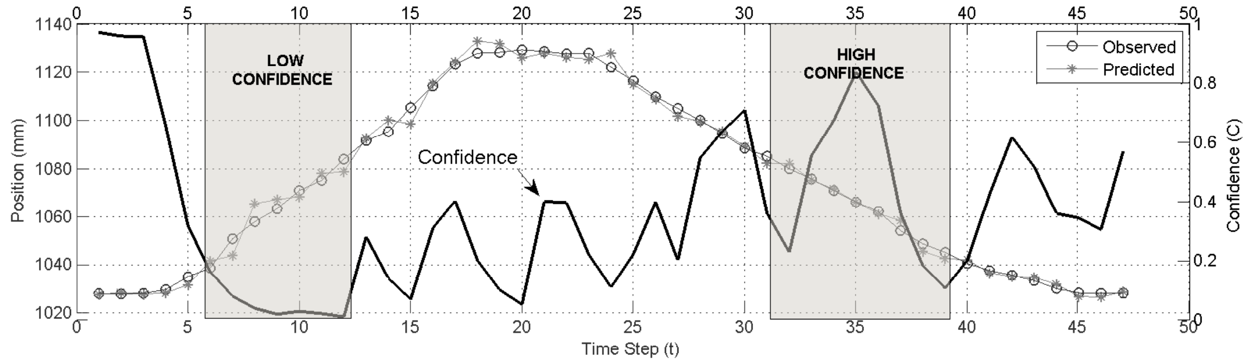

Fig. 5. Confidence value with predictions

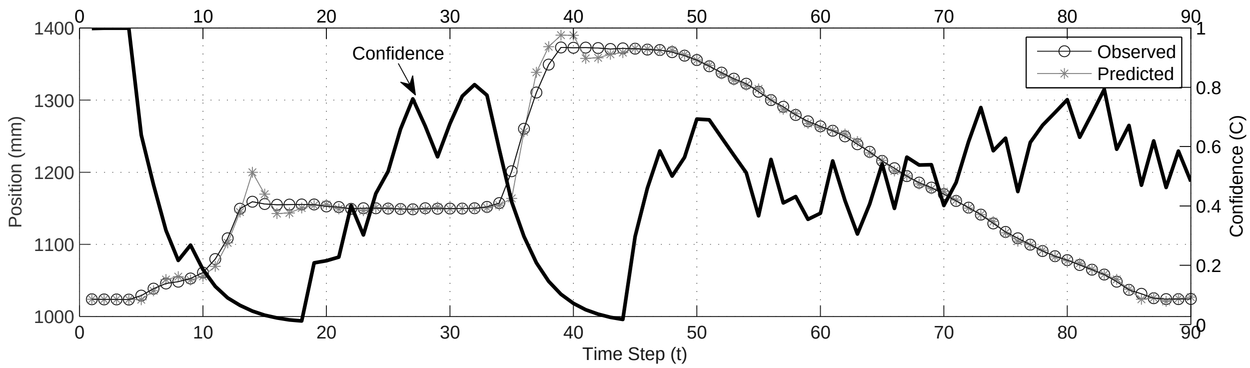

Fig. 6. Irregular case

IV. EXPERIMENTAL RESULTS

In this section we present the experiments performed and the results obtained.

A. Learning the Reactive Controller based on Q - Learning

For Q-learning, a state-action space consisting of 5 states and 5 actions was arbitrarily chosen. The reward r was decided as

$$r = (|Z_2 - Z_1|)_k - (|Z_2 - Z_1)|)_{k+1} \tag{9}$$

where Z_1 and Z_2 represent the position of the human-end and the robot-end of the table respectively.

Hence, if the slant of the table is decreased, the robot receives a positive reward. The action set consists of actions {+2,+1,0,-1,-2}, which correspond to the direction and magnitude of the robot's motion by a defined position step. The position step was set to be 2 cm, since it is the smallest precise movement that can be performed by the robot's arm. Values of the reinforcement learning parameter used were, learning rate $\alpha = 0.9$ and discount factor $\gamma = 0.2$.

10 trials were performed to test how quickly the algorithm could converge to an optimal policy. Median value for number of iterations to converge was 36. The longest episode took 62 iterations before it could converge. Hence, the learning could converge approximately within 40 trials. Each learning trial took about 5 minutes to complete.

B. Prediction

The previously described EKF is used for predicting the human motion one time-step ahead. Each time step is typically about 100 ms, which is the minimum time required for the robot's arm to move from one position to another. Figure 4 shows the predicted and observed values of position, velocity and acceleration.

The predicted position is calculated from (5). True velocity and acceleration are derived from the observed position, and are shown in the figure for comparison with the predicted values.

It can also be observed from fig. 4 that the predictions are inaccurate during the initial steps of the motion. After about 10 time steps the estimates improve. It can be seen that the difference between the predicted velocity and the calculated velocity is very small. The calculated acceleration nearly remains centered at 0 with small changes. This partly justifies our assumption that the acceleration remains nearly constant.

C. Confidence Measure

Figure 5 shows how the confidence ($\mathcal{C}$) of the prediction varies throughout the task, along with the position predictions and observations.

It can be seen that initially, when the task has not begun and the table is still, the predictor accurately estimates the motion to be zero which causes the high confidence value at the beginning. Once the trial starts, in the initial steps,

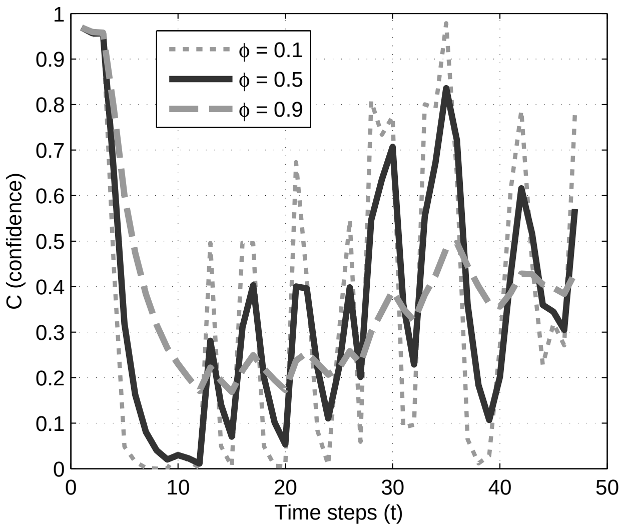

Fig. 7. Effect of the forgetting factor on confidence

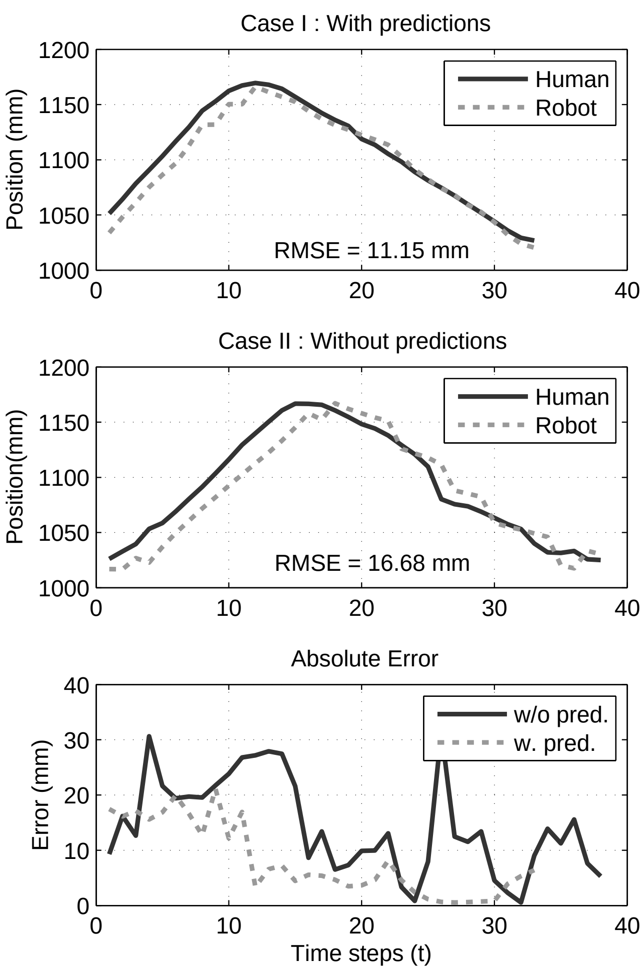

Fig. 8. Comparison

TABLE I
AVERAGE RMSE

Subject	Avg. RMSE w/o Prediction (mm)	Avg. RMSE with Prediction (mm)
1	19.139	12.967
2	23.567	16.591
3	24.872	18.418
4	20.085	15.391
5	22.432	17.684

the predictions are inaccurate because of the drastic change in the motion model. This causes confidence value to drop down suddenly. The reactive controller of the robot becomes dominant in this region. As the predictor gains knowledge about the motion, the predictions go on improving. As the predictions improve, the confidence values also improves. As a result the proactive behavior becomes more dominant.

Figure 7 shows the role of the forgetting factor ϕ in determining the confidence. Since it is not possible to reproduce the exact same trajectory during the task, the confidence trajectories shown in fig. 7 are computed offline step-by-step using data collected from a table lifting task. As seen from (7), a low value of ϕ means that the predictor allots a small weight to older likelihood estimates. Thus, $\mathcal{C}$ mostly depends upon the recently observed $\mathcal{L}$. Hence, if the likelihood values $\mathcal{L}$ change quickly, it causes $\mathcal{C}$ fluctuate heavily. Using a similar reasoning, a large value of ϕ causes the confidence measure to settle very slowly. Hence the robot cannot adapt to the motion changes quickly and generates high confidence values even for incorrect predictions. A good value for ϕ which gives a good tradeoff between smooth variation and adaptability for $\mathcal{C}$ was found to be 0.45.

D. Handling Irregular Cases

One of the major improvements our system offers over most existing systems, is that, no assumption has been made regarding the trajectory of the entire motion. The human has the right to change the trajectory at any point of time, during the trial. Figure 6 shows a case where the motion of the human is not typical. Instead of lifting up the table and keeping it down continuously, the human chooses to take a pause while lifting the table up. Because of this, an abrupt change of motion can be seen around time-step 15. The confidence value drops to zero in 3-4 time steps. During this phase, the robot starts behaving as the follower and simply tries to make the table horizontal using the reactive controller. As the human continues to keep still, the predictor learns this model and predicts zero movement. Hence, although the confidence is high and the robot is the leader, there is no proactive action since the predicted change in position is zero. Again at time-step 35, the human starts moving the table upwards. Again, the robots switches from leader to follower based on the confidence value. Once the motion has been stabilized the robot maintains a confidence value centered somewhere around 0.5.

E. Overall System Performance

In this experiment we evaluate the improvement offered by our system for the table lifting task. If Z_{1t} is the position of human side of the table and Z_{2t} is the position of robot side at any instant t, then the objective is to minimize the absolute error given by

$$AbsoluteError = \sum_t |Z_{1t} - Z_{2t}| \quad (10)$$

We use the motion capture system to record the trajectories of the human and robot table ends. Figure 8 shows these trajectories for cases where the the proposed system was used (case I : with predictions) and the case where only the reactive controller was used (case II : without predictions) The figure also shows the absolute error calculated for the two cases. We use the root mean square error (RMSE) to characterize the performance.

The following observations can be made from fig. 8

- The RMSE for the case I is less than RMSE for case II.
- The motion observed for case I is smoother than that of case II.
- The absolute error is lower in case I.

Quantitative results are provided in table I for multiple users. 5 human subjects were asked to participate in the table lifting task with the robot, one at a time. Each person was asked to lift up the table to a random height and keep it down for 10 trials. Totally, for both the cases, 100 trials were acquired. The table shows the average RMSE for the 10 trials observed for each subject, for each case. It can be seen that, for all the users, RMSE is lower when the proposed approach is used as opposed to a simple reactive approach. Hence a definite improvement can be observed.

V. DISCUSSIONS AND CONCLUSIONS

This section concludes the paper with discussions and future works.

A. Discussions

Figure 9 shows the performance of two humans performing the table lifting task. For the sake of comparison with the human-robot team, the RMSE observed was 6.531 mm.

The motion of the robot is jerky when its reactive behavior is dominant, because of the fixed step sizes. The design of our system is such that the prediction accuracy influences the confidence of prediction. Because of this, many interesting possibilities follow. Better predictions result in better confidence values which allows for proactive robot behavior. Hence, if the human keeps moving smoothly as the robot expects him to move, the motion of the robot is also smooth. This in-turn causes smoother motion of the table as a whole and hence smoother motion of the human, thus resulting in better predictions. However, if the motion of the human is jerky, then the robot is unable to estimate the motion accurately, and hence does not allow for leader behavior. The predictions are not fully utilized in such cases and reflects poor performance. Thus, the results are not only influenced by the robot's performance alone, but also by the human performance. Especially, subject 1 had been working with the system for a longer time than others. Hence, the results for subject 1 were better compared to other human subjects.

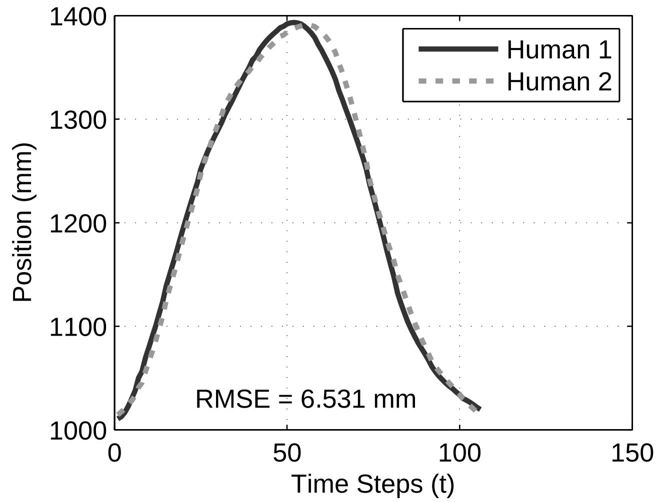

Fig. 9. Human-Human team lifting the table

In fig. 8 we could observe in case I, the trajectory is much smoother when the human is placing the table down as compared to moving upwards. This is because, inherently, the robots motion while lifting the table against gravity is jerky because of the internal control characteristics. This induces some jerks in the human motion also since they are coupled by the table. Because of this, the prediction suffers, which causes lower confidence levels. But while moving downwards, the robot is able to move very smoothly which allows the human to move smoothly and hence the system is utilized to its full potential resulting in better performance. It can also be speculated that sophisticated velocity or torque controlled robots would yield smoother motions and offer better improvements in performance using the proposed technique.

Due to the limitation in the control speed of robot, we could obtain atmost 10 motion capture samples per second. With a faster robot, more samples could be obtained per second which would improve the quality of predictions.

Finally, our work can also be easily extended to proactive teleoperation. The teleoperated robot can choose to take a proactive action based on the confidence values which could reduce the effect of time delays observed in teleoperation and increase transparency.

B. Conclusions

This work contributes to a framework that utilizes human motion prediction to adjust the leader/follower role of the collaborating robot in a co-operative manipulation task. The framework consists mainly of the reactive and proactive controllers. The proactive controller is based on an EKF for human motion prediction. A novel technique to derive a measure of confidence of the prediction has also been proposed. Experimental results were presented to provide conclusive evidence that the proposed approach offers a definite improvement over simple reactive approaches. Additionally, the system does not make any assumptions about the motion trajectory of the object which is practically desirable.

For future works, we propose to utilize longer term predictions. A general case where the human action does not necessarily translate directly to robot action can be considered. Complex objectives in the cooperative task could also be added.

VI. Acknowledgments

This project is partially supported by the NSF grant CISE/CNS 0916864 and CISE/CNS MRI 0923238.

References

[1] N. Hogan, "Impedance control: An approach to manipulation," in *American Control Conference, 1984*, 1984, pp. 304 –313.

[2] M. Rahman, R. Ikeura, and K. Mizutani, "Investigating the impedance characteristic of human arm for development of robots to co-operate with human operators," in *Systems, Man, and Cybernetics, 1999. IEEE SMC '99 Conference Proceedings. 1999 IEEE International Conference on*, 1999.

[3] Y. Maeda, T. Hara, and T. Arai, "Human-robot cooperative manipulation with motion estimation," in *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*, 2001.

[4] T. Flash and N. Hogan, "The coordination of arm movements: an experimentally confirmed mathematical model," *The Journal of neuroscience*, vol. 5, no. 7, pp. 1688–1703.

[5] B. Corteville, E. Aertbelien, H. Bruyninckx, J. De Schutter, and H. Van Brussel, "Human-inspired robot assistant for fast point-to-point movements," in *Robotics and Automation, 2007 IEEE International Conference on*, 2007, pp. 3639 –3644.

[6] P. Evrard and A. Kheddar, "Homotopy switching model for dyad haptic interaction in physical collaborative tasks," in *EuroHaptics conference, 2009 and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. World Haptics 2009. Third Joint*, 2009, pp. 45 –50.

[7] C. Smith and P. Jensfelt, "A predictor for operator input for time-delayed teleoperation," *Mechatronics*, vol. 20, no. 7, pp. 778 – 786, 2010, special Issue on Design and Control Methodologies in Telerobotics.

[8] C. Passenberg, A. Peer, and M. Buss, "A survey of environment-, operator-, and task-adapted controllers for teleoperation systems," *Mechatronics*, vol. 20, no. 7, pp. 787 – 801, 2010, special Issue on Design and Control Methodologies in Telerobotics.

[9] N. Jarrasse, J. Paik, V. Pasqui, and G. Morel, "How can human motion prediction increase transparency?" in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, May 2008, pp. 2134 –2139.

[10] S. Clarke, G. Schillhuber, M. Zaeh, and H. Ulbrich, "Prediction-based methods for teleoperation across delayed networks," *Multimedia Systems*, vol. 13, pp. 253–261, 2008, 10.1007/s00530-007-0103-z.

[11] S. Miossec and A. Kheddar, "Human motion in cooperative tasks: Moving object case study," in *Robotics and Biomimetics, 2008. ROBIO 2008. IEEE International Conference on*, 2009, pp. 1509 –1514.

[12] E. Gribovskaya, A. Kheddar, and A. Billard, "Motion Learning and Adaptive Impedance for Robot Control during Physical Interaction with Humans," in *Proceedings of IEEE International Conference on Robotics and Automation*, 2011.

[13] D. Simon, D. L. Simon, and D. Viassolo, "Kalman filter constraint switching for turbofan engine health estimation. Discussion," *European journal of control*, vol. 12, no. 3, pp. 331–345.