Learning Human-like Hand Reaching for Human-Robot Handshaking

Vignesh Prasad\textsuperscript{1}, Ruth Stock-Homburg\textsuperscript{1}, Jan Peters\textsuperscript{1,2}
\{vignesh.prasad, ruth.stock-homburg, jan.peters\}@tu-darmstadt.de

Abstract—One of the first and foremost non-verbal interactions that humans perform is a handshake. It has an impact on first impressions as touch can convey complex emotions. This makes handshaking an important skill for the repertoire of a social robot. In this paper, we present a novel framework for learning human-robot handshaking behaviours for humanoid robots solely using third-person human-human interaction data. This is especially useful for non-backdrivable robots that cannot be taught by demonstrations via kinesthetic teaching. Our approach can be easily executed on different humanoid robots. This removes the need for re-training, which is especially tedious when training with human-interaction partners. We show this by applying the learnt behaviours on two different humanoid robots with similar degrees of freedom but different shapes and control limits.

I. INTRODUCTION

Physical contact, especially instantaneous contact is of great importance in various human-robot interaction (HRI) applications\textsuperscript{[1]}, especially since touch conveys information about the emotional state of a person\textsuperscript{[2], [3]}. This enables a special kind of emotional connection to human users during the interaction\textsuperscript{[4], [5]}. Among such interactions, handshaking is a simple, natural interaction that is used in many social contexts\textsuperscript{[6], [7]}. It plays an important role in shaping first impressions\textsuperscript{[6], [7]} as it is usually the first non-verbal interaction that takes place in a social context.

Thus, having a good handshake would not only widen the expressive abilities of a social robot but also provide a strong first impression for further interactions and is, therefore, an important skill required for the acceptance of social robots.

The human-likeness of robotic motions is an important aspect as movements can exaggerate feelings of uncanniness felt towards humanoid robots\textsuperscript{[8]} as compared to their static appearance. Accordingly, the robot must be able to detect and predict the motion of the human body and react within a reasonable time. Otherwise, the physical interaction would be slow and unnatural. Vinayavekhin et al.\textsuperscript{[9]} aim to bridge this gap in the context of human-robot handshaking using Long Short Term Memory networks (LSTMs) for predicting the human-hand motion and devise a simple controller for the robot’s response motion. Such interaction dynamics during human-robot handshaking are also captured implicitly by Campbell et al.\textsuperscript{[10]} who learn a joint distribution over the trajectories of the human and the robot. However, their approach is robot-specific and would need to be re-trained with human interaction partners when applied to new robots.

\textsuperscript{1}Technical University of Darmstadt, Germany
\textsuperscript{2}Max Planck Institute for Intelligent Systems, Tübingen, Germany

Fig. 1: We propose a framework to learn human-robot handshaking from human-human interactions. We first learn to predict the human’s motions using a LSTM network and then learn a robot controller in the form of Probabilistic Movement Primitives (ProMPs) that can be conditioned in real-time using the predicted motions. Both are learnt just from skeleton data of human handshaking interactions, in a robot-agnostic fashion by leveraging the similarities between the degrees of freedom of humans and humanoid robots.

This can be more tedious than kinesthetic teaching for traditional robotic tasks. To some extent, this can be circumvented by learning from end-effector trajectories instead, like in\textsuperscript{[11]}. They use Deep Reinforcement Learning with a human imitation reward to learn suitable motions but do not look at the interactive nature of the task. A detailed survey on human-robot handshaking can be found in\textsuperscript{[12]}.

To make way for a more adaptable method, we propose a framework that can be easily transferred across different humanoid robots without the need for re-training. We build on the framework proposed in\textsuperscript{[9]} by improving the robot controller such that it learns from human-human interactions, that can be seen in Fig.\textsuperscript{[1]} Moreover, in the case of pneumatically controlled robots which are not backdrivable and hence cannot be kinesthetically taught it is imperative to explore new ways of learning robot motions.

Jindai and Watanabe\textsuperscript{[13]} observed that hand motions of interaction partners are very similar in human-human handshaking. Their data shows curved trajectories, similar to those shown in Fig.\textsuperscript{[5]} Instead, Vinayavekhin et al.\textsuperscript{[9]} propose simple straight-line trajectories to the detected hand location. In this regard, we learn directly from the joint motions of human-human interactions, similar to tele-operating a robot\textsuperscript{[14]}. We learn a distribution over the extracted joint angle trajectories in the form of Probabilistic
Motion Primitives (ProMPs) [15]. We exploit an important property of ProMPs which is that they can be conditioned to reach a particular location in their task space [16]. We follow the LSTM-based motion prediction proposed in [9] to estimate the human hand’s final position and condition the ProMPs to meet the human hand at the grasping point. We do so in a way that can seamlessly be executed on different humanoid/android robots, that have similar degrees of freedom as a human. Our main contribution is a principled pipeline that learns interactive actions directly from human demonstrations, removing the need to train robots with human interaction partners for such tasks. Since we learn the human joint motions, the learned behaviour can be easily transferred across different humanoid robots that have similar degrees of freedom. We show this by applying the learned behaviour on two humanoid robots, both of which are different in sizes and control limits.

II. Preliminaries

In this section, we provide a brief introduction of the methods which we use in our approach, namely LSTMs in Sec. II-A and ProMPs in Sec. II-B.

A. Long Short Term Memory

Long Short Term Memory (LSTM) networks [17] are a special kind of recurrent neural network (RNN) architecture for learning long term dependencies in time sequences. Typical RNNs just propagate the information forward from the previous timestep and combine it with the current timestep to get a prediction. LSTMs have multiple gates that decide how much information from previous time steps needs to be retained and how much from the current input would be used for the prediction. This allows them to learn long term correlations in temporal sequences, which makes them an ideal candidate for human motion prediction (a survey can be found at [13]). Moreover, we use the entire upper body 3D joints instead of just the arm or hand since body language can help in gauging the emotional state of a person [19], [20]. LSTMs have also shown good performance in this regard [20], which could be integrated into our approach in the future to predict emotional cues for handshaking as well.

B. Probabilistic Movement Primitives

Probabilistic Movement Primitives (ProMPs) [15] are a framework for learning distributions over robot trajectories. Given a set of trajectories \( \tau = [y_1 \ldots y_T] \), where each \( y_i \) is a vector of joint angles or positions, velocities etc., the trajectory distribution is parameterized as: \( p(y_i|\omega) = \mathcal{N}(y_i|\Psi(t)^T \omega, \Sigma_y) \) where \( \Psi(t) \) is a diagonal matrix of time dependent basis functions and \( \Sigma_y \) is a gaussian noise.

Variations in the trajectories are modelled by sampling \( \omega \) from a prior \( \omega \sim \mathcal{N}(\mu_{\omega}, \Sigma_{\omega}) \). The likelihood can now be written in terms of the parameters \( \theta_{\omega} = \{\mu_{\omega}, \Sigma_{\omega}\} \) as

\[
p(y_i|\theta_{\omega}) = \int \mathcal{N}(y_i|\Psi(t)^T \omega, \Sigma_y) \mathcal{N}(\omega|\mu_{\omega}, \Sigma_{\omega}) d\omega
\]

The parameters \( \mu_{\omega}, \Sigma_{\omega} \) are learnt by optimizing Eq. (1).

To counter the different speeds of executions in the demonstrations, a phase variable is used \( z_i = \frac{t_i - t}{T} \) which is a normalized version of the actual trajectory time. This allows us to control the speed of execution while testing as well.

One important characteristic of ProMPs, that we exploit in this work, is the ability to condition them with particular observation(s) and applying Bayes rule to optimize the likelihood of the given observation(s). This can be done in two ways: joint space conditioning and task space conditioning.

In joint space conditioning, we wish to reach a given joint configuration \( y^*_t \) at time \( t \). This is done by applying Bayes theorem at the given time with the observation \( \{y^*_t, \Sigma^*_y\} \):

\[
p(\omega|y^*_t, \Sigma^*_y) \propto \mathcal{N}(y^*_t|\Psi(t)^T \omega, \Sigma_y) p(\omega) \tag{2}
\]

which yields a new distribution for \( \omega \) defined as:

\[
K = \Sigma_{\omega} \Psi(t)(\Sigma^*_y + \Psi(t)^T \Sigma_{\omega} \Psi(t))^{-1}
\]

\[
\mu^*_{\omega} = \mu_{\omega} + K(y^*_t - \Psi(t)^T \mu_{\omega}) \tag{3}
\]

\[
\Sigma^*_{\omega} = \Sigma_{\omega} - K\Psi(t)^T \Sigma_{\omega} \tag{4}
\]

In task space conditioning, given a target 3D location \( x^*_t \sim \mathcal{N}(\mu_{x}, \Sigma_x) \) at time \( t \), a joint space configuration \( y_t \) is estimated such that it maximises \( p(y_t | x^*_t, \theta_{\omega}) \) to reach \( x_t \) while staying close to the learnt ProMP \( \theta_{\omega} \). Further details about task-space conditioning can be found in [16].

III. Proposed Approach

In this section, we provide a detailed explanation of our novel approach of learning robot-agnostic handshaking from human handshaking interactions using LSTMs for explicitly encoding the interaction dynamics and using ProMPs for learning robot control from human skeleton data. This is especially useful when one of the robots is non-backdrivable and cannot be taught via kinesthetic teaching. An overview of this novel framework can be seen in Fig. 2 and is explained in Sec. III-A. The hand location prediction for enabling the "interactiveness" of the method is explained in Sec. III-B. Learning ProMPs from the human skeleton data for humanoid robots is explained in Sec. III-C. The combination of the hand prediction and the subsequent conditioning of the ProMP with the predicted location is explained in Sec. III-D.

A. Overview

Given skeleton data from human-human interactions, we use the 3D upper body joint locations to train the LSTM network to predict the final hand location. Learning the ProMP can be thought of as teaching the robot via teleoperation. We extract joint angles from the skeleton data which are similar as the degrees of freedom of the humanoid robots (shoulder yaw, pitch, roll and elbow angle) with which we learn the ProMP over the extracted joint angle trajectories. This circumvents the need for kinesthetic teaching which is not possible for non-backdrivable robots like one of the robots used in this work. During testing, the human interaction partner’s skeleton is tracked by an external RGB-D camera. This is given to the LSTM network to predict a target hand location. Given the predicted and the true hand
Fig. 2: Overview of our proposed method. We extract the human skeleton which is given to an LSTM network to predict the final hand position. Using the predicted and actual position of the hand, a target location is inferred based on the time lapsed during the interaction. These target locations are used to condition the learnt ProMP controller so that the robot hand meets the human hand at the final position.

B. Predicting Human Hand Motions using LSTMs

During handshaking and other interactive behaviours, we as humans try to predict where the interaction partner’s hand would go based on the motion and body language and move accordingly, adapting the motion to meet the hand at the final location as the interaction progresses. To explicitly encode this kind of predictive behaviour, we train an LSTM using the upper body skeleton joints from the human-human interaction sequences. We use the same architecture as in [9] where given a sequence of $n$ frames of upper body skeleton joints in 3D $b_1, b_2, b_3 \ldots b_n (b_i \in \mathbb{R}^{15 \times 3})$, the final 3D hand location $\hat{h}_n$ is predicted at each time step. The upper body is used since it allows us to learn relations between body language and the motion prediction, instead of using only the hand or arm motions. We prefer this approach over learning the joint trajectories of both the human and the robot since this can be explicitly used for any robot that the human interacts with, rather than learning robot-specific interactions, which may not be as easy to transfer to a different robot.

C. Learning from Human Motions using ProMPs

Given the 3D skeleton data of a human, the shoulder angles (roll, pitch, yaw) and the elbow angle are extracted for the right hand (details can be found in [14]), which can be seen as learning by teleoperation. We use the joint angles instead of the 3D joint locations as the angles capture the underlying control that we as humans perform and more importantly, this is similar to the degrees of freedom present for the humanoid robots that we use in this work, as shown in Fig. 3. Since the wrist angles are difficult to compute due to errors in the tracking, we defer this to our future work.

Given the extracted joint angle trajectories, the ProMP weights are computed by optimizing Eq. 1 using ridge regression as $\omega_i = (\Psi^T \Psi + \lambda I)^{-1} \Psi^T \tau_i$ where $\tau_i$ is the $i^{th}$ trajectory and $\Psi$ is the corresponding list of basis functions matrices. We set $\lambda = 10^{-10}$ as larger values cause a divergence in the learning. Additionally, as shown in [21], the jerk can be minimized as $\omega_i = (\Psi^T \Psi + \lambda \Gamma^T \Gamma)^{-1} \Psi^T \tau_i$ where $\Gamma$ is the third derivative of $\Psi$ w.r.t time. From this, we can calculate the ProMP parameters $\mu_\omega, \Sigma_\omega$ as the mean and covariance of the estimated weight vectors.

D. ProMP Conditioning with Predicted Hand Locations

During testing, the human’s skeleton is tracked and the upper body joints are given to the LSTM network to predict a final hand location. Like in [9], to shift to the human hand towards the end of the trajectory, thereby converging to the
true hand position, the target position at the given time step \( h_t^* \) for conditioning the ProMP is calculated as:
\[
h_t^* = (1 - \sigma(t + \alpha)) \hat{h}_t + \sigma(t + \alpha) h_t
\]
where \( \hat{h}_t \) is the predicted hand location, \( h_t \) is the tracked hand location, \( \sigma(\cdot) \) is the sigmoid function and \( \alpha \) is constant to center the sigmoid at half the trajectory. This ensures a smooth transition between the predicted and tracked hand location towards the end of the interaction.

Given the target hand location, the ProMP is conditioned by maximising \( p(y_t^* | x_t^*, \theta) \) where \( x_t^* = h_t^* \). This boils down to the following optimization problem\(^{16}\):
\[
y_t^* = \arg\min_y \lambda_x \| \mu_x - f(y) \|^2 + \lambda_y \| \mu_y - y \|^2
\]
where \( f(y) \) is the forward kinematics to estimate the end-effector position given joint angles \( y \), \( \mu_y = \Psi(t)^T \mu_w \) and \( \Sigma_y = \Psi(t)^T \Sigma_w \Psi(t) \) are the marginal distribution of the learnt ProMP and \( \lambda_x \) and \( \lambda_y \) are weights that penalize the deviations from the target position and from the ProMP respectively. Given \( y_t^* \), we perform joint space conditioning to obtain the new ProMP parameters (Eqs. 3 - 5).

\[\text{Fig. 4: Humanoid robots used in this work}\]

**IV. EXPERIMENTS AND RESULTS**

In this section, we introduce the robots and go into further detail about the implementation details of our method (Sec. \[IV-A\]), the dataset used to train the human motion prediction network and the robot-agnostic reaching behaviour (Sec. \[IV-B\]) and the results of the LSTM network predictions and reaching behaviours on different robots (Sec. \[IV-C\]).

**A. Experimental Setup**

We use two humanoid robots in our experiments. The first is Pepper (Fig. 4a), a humanoid social robot from Aldebaran Robotics\(^22\), having 6 degrees of freedom in each arm. The second is a custom-made pneumatically controlled android robot, Elenoide (Fig. 4b), having 14 degrees of freedom per arm. We use NuiTrack\(^23\) on data captured by an Intel RealSense\(^24\) to track the human partner’s skeleton. The entire system is controlled using ROS\(^25\).

The hand prediction network is implemented using PyTorch\(^26\) and consists of 2 LSTM layers with a hidden dimension of 64 followed by a fully connected layer that predicts the final 3D hand location at each time-step. The network was trained with a batch size of 32 for 200 epochs using the Adam optimizer\(^27\). Given that Eq. 6 needs an estimate of the trajectory length, we use an estimated trajectory length of 32 frames (\(~ 1\) sec.), which is the median length of the training trajectories. For learning the ProMP, 3 RBF kernels with equally spaced centers and a scale of 0.01 are used. The ProMPs are implemented using a modified version\(^1\) of the Bayesian Interaction Primitives Framework\(^28\). We use SciPy\(^29\) for the least-squares estimation of the ProMP weights and for solving the inverse kinematics during task space conditioning (Eq. 7), with \( \lambda_x = \lambda_y = 0.5 \).

**B. Dataset**

We use the skeleton data from the handshaking interactions present in the NTU RGB+D dataset\(^30\) for training the hand prediction network and for learning the ProMPs. The skeleton data is recorded using a Microsoft Kinect\(^31\) v2, which provides as a set of 25 joints for each person present in the video, of which we use the 15 upper body joints. Fig. 5 shows some sample hand trajectories of one of the interaction partners from the handshaking interactions. The darker shade denotes the starting of the trajectories and the green colour denotes the ending of the trajectories.

\[\text{Fig. 5: Sample hand trajectories for handshaking from the NTU RGB+D dataset. (dark - starting, bright - ending)}\]

From the skeletons of the handshaking action, we remove those in which the handshaking is done with the left hand. We further remove those which have errors or discontinuities in the skeleton tracking. Among those remaining, we filter out the part of the trajectory after the hands are grasped and the initial part of the trajectory where no movement is present. Of these, we use 500 trajectories, randomly split into 400 training and 100 testing trajectories to train the hand prediction network. We further filter out trajectories that still have irregularities and use the remaining 197 trajectories and extract the joint angles of the right hand from each of the trajectories.

\[\text{https://github.com/souljaboy764/intprim}\]
Fig. 6: Hand Position Prediction Example. The leftmost frame is the starting frame, the rightmost is the final frame, the others are equally spaced between them. The blue dots denote the upper body skeleton joints, the red dot is the predicted final location from the network and the green dot is the target location calculated using Eq. 6.

trajectories to train the ProMP. Further details about the joint angle extraction can be found in [14].

C. Results

1) Hand Prediction Accuracy: In Fig. 7 we show the training loss and testing loss as the training progresses. As it can be seen, the network predictions are accurate within a few centimetres, which can be seen in Fig. 6. This is not an issue since the prediction is only used in the initial stages of the interaction. The target location calculated using Eq. 6 switches smoothly to the true tracked location as the trajectory comes towards the end. This is also shown in Fig. 8 where the target trajectory (green) can be seen shifting from the network prediction (red) to the true location (blue).

Fig. 7: Accuracy of the hand prediction network.

2) Hand Reaching Results: Given that we explicitly condition the learnt ProMP on a target 3D location, we attain a good accuracy with the hand reaching behaviours. An example interaction can be seen in Fig. 9 where the timeliness of the interaction can also be seen as the robot starts moving as it senses the human hand moving due to the explicit modelling of the interactive behaviour. Few qualitative samples showing the spatial robustness of our method can be seen in Fig. 10. We obtain an average error of $0.062 \pm 0.025$ m of the final hand location and the robot end-effector for Pepper over 10 interaction trajectories. Some errors are attributed to errors in the hand detection and calibration of the setup as well. Due to improper feedback from the pneumatic controller for Elenoide, the average reaching error cannot be properly calculated.

V. CONCLUSION AND FUTURE WORK

In our paper, we extend the framework proposed by Vinayavekhin et al. [9] for human-robot handshaking, by learning a robot controller from human interactions. This is especially important when using imitation learning for pneumatically controlled robots that are not backdrivable, like one of the robots used in this paper. The dynamics of the interaction are explicitly modelled using an LSTM network, whose output is used to condition the robot controller to meet the human hand at the predicted location. It does so in a timely and smooth manner. We exploit the fact that humanoid robots have similar degrees of freedom as a human arm and apply the learnt ProMP controller to different humanoid robots directly without any re-training, which is especially tedious if the training requires human interaction partners.

Currently, we only look at the reaching phase of handshaking. The grasping and shaking phases requires a suitable synergic mechanism that can sense the forces applied and react accordingly. Further research is also required to learn ProMPs for shaking along with reaching, as one is rhythmic, while the other is stroke-based. Sensing the context and reacting accordingly is also important, for example speeding up the motion or strengthening the grasp, could change the way the interaction is perceived. Finally, the true test of how good an interactive behaviour is, would require trials with human partners, which we currently defer to our future work.
Fig. 9: An example of an interaction generated by our proposed approach. The top row shows the interaction, the second row shows the robot joint angles and the third row shows the end effector location in the robot’s frame. The solid lines denote the observed values, the dashed lines represent the values from the commands generated by the ProMP and the dot at the end represents the target value used to condition the ProMP.

Fig. 10: Spatial robustness of the learnt reaching behaviour for different hand locations with different robots.
