Bütepage, Judith; Ghadirzadeh, Ali; Öztimur Karadağ, Özge; Björkman, Mårten; Kragic, Danica

Imitating by Generating

Published in:
Frontiers in Robotics and AI

DOI:
10.3389/frobt.2020.00047

Published: 16/04/2020

Document Version
Publisher's PDF, also known as Version of record

frontiers in Robotics and AI

ORIGINAL RESEARCH
published: 16 April 2020
doi: 10.3389/frobt.2020.00047

Imitating by Generating: Deep Generative Models for Imitation of Interactive Tasks

Judith Bütepage[1]*, *Ali Ghadirzadeh*[1,2], *Özge Öztimur Karadağ*[1,3], *Mårten Björkman*[1] and *Danica Kragic*[1]

[1] Robotics, Perception and Learning, School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, Stockholm, Sweden, [2] Intelligent Robotics Research Group, Aalto University, Espoo, Finland, [3] Department of Computer Engineering, Alanya Alaaddin Keykubat University, Antalya, Turkey

To coordinate actions with an interaction partner requires a constant exchange of sensorimotor signals. Humans acquire these skills in infancy and early childhood mostly by imitation learning and active engagement with a skilled partner. They require the ability to predict and adapt to one's partner during an interaction. In this work we want to explore these ideas in a human-robot interaction setting in which a robot is required to learn interactive tasks from a combination of observational and kinesthetic learning. To this end, we propose a deep learning framework consisting of a number of components for (1) human and robot motion embedding, (2) motion prediction of the human partner, and (3) generation of robot joint trajectories matching the human motion. As long-term motion prediction methods often suffer from the problem of regression to the mean, our technical contribution here is a novel probabilistic latent variable model which does not predict in joint space but in latent space. To test the proposed method, we collect human-human interaction data and human-robot interaction data of four interactive tasks "hand-shake," "hand-wave," "parachute fist-bump," and "rocket fist-bump." We demonstrate experimentally the importance of predictive and adaptive components as well as low-level abstractions to successfully learn to imitate human behavior in interactive social tasks.

Keywords: imitation learning, human-robot interaction, generative models, deep learning, sensorimotor coordination, variational autoencoders

OPEN ACCESS

Edited by:
Séverin Lemaignan,
Bristol Robotics Laboratory,
United Kingdom

Reviewed by:
Soheil Keshmiri,
Advanced Telecommunications Research Institute International (ATR), Japan
Chenguang Yang,
University of the West of England, United Kingdom

***Correspondence:**
Judith Bütepage
butepage@kth.se

Specialty section:
This article was submitted to Human-Robot Interaction, a section of the journal Frontiers in Robotics and AI

Received: *31 October 2019*
Accepted: *17 March 2020*
Published: *16 April 2020*

Citation:
Bütepage J, Ghadirzadeh A, Öztimur Karadağ Ö, Björkman M and Kragic D (2020) Imitating by Generating: Deep Generative Models for Imitation of Interactive Tasks. Front. Robot. AI 7:47. doi: 10.3389/frobt.2020.00047

1. INTRODUCTION

Physical human-robot interaction requires the robot to actively engage in joint action with human partners. In this work, we are interested in robotic learning of physical human-robot tasks which require coordinated actions. We take inspiration from psychological and biological research and investigate how observational and kinesthetic learning can be combined to learn specific coordinated actions, namely interactive greeting gestures.

In a more general context, coordinated actions between humans can be of functional nature, such as handing over an object, or of social importance, such as shaking hands as a greeting gesture. Thus, joint actions encompass any kind of coordination of actions in space and time in a social context. In general, joint actions require the ability to share representations, to predict others' actions and to integrate these predictions into action planning (Sebanz et al., 2006). On a sensorimotor level coordinated actions require a constant coupling between the partners' sensory

and motor channels (Vesper et al., 2017). We aim at making use of sensorimotor patterns to enable a robot to engage with a human partner in actions that require a high degree of coordination such as hand-shaking.

The acquisition of the ability to engage in joint action during human infancy and early childhood is an active field of research in psychology (Brownell, 2011). Interaction is mostly learned in interaction, from observation, active participation, or explicit teaching. While cultural differences exist, children are commonly presented with the opportunity to learn through guided participation in joint action with more experienced interacting partners (Rogoff et al., 1993). In the robotics community two prominent techniques to learn actions from others are *learning from demonstration* and *imitation learning* (Billard et al., 2008; Osa et al., 2018). Learning from demonstration can be seen as a form of imitation learning. It requires a teacher to intentionally demonstrate to a learner how an action should be performed. In a robotic learning scenario, this can imply direct kinesthetic teaching of trajectories. General imitation learning on the other hand includes also learners who passively observe an action and replicate it without supervision. When observing a human, who often has a different set of degrees of freedom, the robotic system first needs to acquire a mapping between embodiments before a motion can be imitated (Alissandrakis et al., 2007).

In this work, we are interested in teaching a robot to coordinate with a human in time and space. Therefore, we require adaptive and predictive models of sensorimotor patterns such as joint trajectories and motor commands of interactive tasks. To this end, we develop deep generative models that represent joint distributions over all relevant variables over time. The temporal latent variables in these models encode the underlying dynamics of the task and allow for a sensorimortor coupling of the human and the robot partner. As depicted in **Figure 1**, collecting data by kinesthetic teaching for human-robot interaction tasks is tedious and time-consuming. We propose to first model the dynamics of human-human interaction and subsequently use the learned representation to guide the robot's action selection during human-robot interaction.

Before diving into the theory, in the next section we will shortly introduce the field of robotic imitation learning and point out how the general field differs from the requirements needed for imitation learning for interaction. Finally, we will motivate our choice of model and explain the basic assumptions of deep generative models.

2. BACKGROUND

Traditionally, robotic imitation learning is applied to individual tasks in which the robot has to acquire e.g., motor skills and models of the environment. Our goal is to extend these ideas to interactive settings in which a human partner has to participate in action selection. Thus, we aim at transferring knowledge form observing human-human interaction (HHI) to human-robot interaction (HRI).

2.1. Robotic Imitation Learning of Trajectories

Imitation learning is concerned with acquiring a policy, i.e., a function that generates the optimal action given an observed state. While reinforcement learning usually solves this task with help of active exploration by the learning agent, in imitation learning the agent is provided with observations of states and actions from which to learn. These demonstrations can either be generated in the agent's own state space, e.g., by tele-operation (Argall et al., 2009), or in the demonstrators embodiment, e.g., a human demonstrating actions for a robot. In this work we combine these approaches to teach a robot arm trajectories required for a number of interactive tasks.

Learning trajectory generating policies from demonstration has been addressed with for example a combination of Gaussian Mixture Models and Hidden Markov Models (Calinon et al., 2010), probabilistic flow tubes (Dong and Williams, 2011, 2012), or probabilistic motion primitives (Maeda et al., 2017b). The general strategy in this case is to first gather training data in the form of trajectories and to align these temporally e.g., with the help of Dynamic Time Warping (Sakoe and Chiba, 1978). Once the training data has been pre-processed in this way, the model of choice is trained to predict the trajectory of robotic motion for a certain task. During employment of the model, the online trajectory needs to be aligned with the temporal dynamics of the training samples in order to generate accurate movements. Depending on the trajectory representation, e.g., torque commands or Cartesian coordinates, the model's predictions might be highly dependent on the training data. For example, when the task is to learn to grasp an object at a certain location, the model might not generalize to grasping the same object at a different location.

This constant need of alignment and reliance on demonstrations hampers the models to work in a dynamic environment with varying task demands. For example, if the task is to shake hands with a human, the number of shaking cycles and the length of each individual shake can vary from trial to trial and have to be estimated online instead of being predicted once prior to the motion onset. These requirements for online interaction are discussed in more detail below.

2.2. Requirements for Online Interaction

Interaction with humans requires a robotic system to be flexible and adaptive (Dautenhahn, 2007; Maeda et al., 2017a). To meet these requirements, the robot needs to be able to anticipate future human actions and movements (Koppula and Saxena, 2015). Thus, imitation learning for interaction is different from non-social imitation learning as it requires to learn a function not only of one's own behavior, but also of the partner's behavior.

These requirements stand therefore in contrast to the approaches to imitation learning discussed in section 2.1 which focus on learning a trajectory of a fixed size. Maeda et al. (2017a) address the problem of adjusting to the speed of the human's actions by introducing an additional phase variable. This variable can be interpreted as an indication of the progress of the movement of the human to which the robot has to adapt.

FIGURE 1 | Kinesthetic teaching of a human-robot hand shake. The human partner is wearing a motion capture suit to record joint positions.

However, such an approach is only feasible for interactions which require little mutual adaptation beyond speed. For example, during a hand-shake interaction, it is not only important to meet the partner's hand at an appropriate time, but also to adjust to the frequency and height of every up-and-down movement. Thus, online interaction requires the prediction of the partner's future movements in order to adapt to them quickly and a continuous update of these predictions based on sensory feedback.

An additional requirement for natural human-robot interaction is to provide precise coordination. We envision a robot to actively engage in an interaction such that the human partner does not have to wait with a stretched arm until the robot reacts and moves its arm to engage in a hand-shake. Making use of predictive models allows the robot to initiate its movements before the human has reached the goal location. These models also provide a basis for collision-free path planning to assure safe interaction in shared workspaces.

Since humans are involved in the data collection process and kinesthetic teaching is time consuming and requires expert knowledge, the amount of training data is restricted. Therefore, any method used to learn trajectories must be data efficient. Many modern imitation learning techniques build on ideas from deep reinforcement learning (Li et al., 2017; Zhang T. et al., 2018) which is often data intensive. We rely on a model class which is regularized by its Bayesian foundation and therefore less prone to overfit to small datasets. This model class of deep latent variable models has been mostly used to model images. Here, we take inspiration from earlier work in which we model human motion trajectories (Bütepage et al., 2018a) and robot actions (Ghadirzadeh et al., 2017) with help of deep generative models. We extend the ideas to represent the dynamics of human-robot interaction in a joint model. For those unfamiliar with the ideas of Variational Autoencoders, we introduce the underlying concept of this model class below.

2.3. Deep Generative Models and Inference Networks

In this work, we model human and robotic motion trajectories with help of Variational Autoencoders (VAEs) (Rezende et al., 2014; Kingma and Welling, 2015), that is a class of deep generative models. In contrast to Generative Adversarial Networks (Goodfellow et al., 2014) and flow-based methods (Dinh et al., 2017; Kingma and Dhariwal, 2018), VAEs allow us to define our assumption in terms of a probabilistic, latent variable model in a principled manner. While we focus on the main concepts and the mathematical foundations of VAEs, we refer the reader to Zhang C. et al. (2018) for an in-depth review on modern advances in variational inference and VAEs. In the next section, we will shortly introduce the concepts of variational inference.

2.3.1. Variational Inference

To begin with, we assume that the observed variable, or data point, $\mathrm{x} \in \mathbb{R}^{d_x}$ depends on latent variables $\mathrm{z} \in \mathbb{R}^{d_z}$. If the dataset consists of images, the latent variables or factors z describe the objects, colors, and decomposition of the image. If, as we will introduce later, the dataset consists of human or robot joint movements, the underlying factors describe the general movement patterns. For example, a waving movement, in which many joints are involved, can be described by a single oscillatory latent variable. The dimension of z is smaller than the dimension of x, i.e., $d_z < d_x$, as it is a compressed representation of the observation. The precise size of the dimension is a modeling choice.

In general, this model describes a joint distribution over both variables $p_\theta(\mathrm{x}, \mathrm{z}) = p_\theta(\mathrm{x}|\mathrm{z})p_\theta(\mathrm{z})$ where θ are parameters. This modeling assumption allows us to generate new observations with help of the mathematical model instead of employing a physical system. First, a latent variable is sampled, from a prior distribution $\mathrm{z} \sim p_\theta(\mathrm{z})$. For example, to generate a waving arm movement, we sample where in the oscillation the arm starts and the initial velocity. Then we sample the actual poses conditioned on these latent variables. The conditional distribution $\mathrm{x} \sim p_\theta(\mathrm{x}|\mathrm{z})$ encodes the mapping from the latent space to the observed space. Thus, the generative process looks as follows:

$$\mathrm{x} \sim p_\theta(\mathrm{x}|\mathrm{z}),\ \mathrm{z} \sim p_\theta(\mathrm{z}). \tag{1}$$

In order to determine the structure of the latent variables that were generated on an observed set consisting of n data points $\mathrm{X} = \{\mathrm{x}_i\}_{i=1:n}$, one requires access to the posterior distribution $p_\theta(\mathrm{z}_i|\mathrm{x}_i)$ for each data point x_i. Often exact inference of this term is intractable which is why one recedes to approximate inference techniques such as Monte Carlo sampling and variational inference (VI). VAEs combine VI for probabilistic models with the representational power of deep neural networks. VI is an optimization based inference technique which estimates the true posterior distribution $p_\theta(\mathrm{Z}|\mathrm{X})$ with a simpler approximate distribution $q_\phi(\mathrm{Z})$ where ϕ are parameters and $\mathrm{Z} = \{\mathrm{z}_i\}_{i=1:n}$ is the set of latent variables corresponding to the data set. A common approach is the mean-field approximation which assumes that the latent variables are independent of each other

$q_\phi(\mathrm{Z}) = \prod_{i=1}^{n} q_\phi(\mathrm{z}_i)$. As an example, if $q_\phi(\mathrm{z}_i)$ follows a Gaussian distribution, we need to identify a mean μ_i and variance σ_i for every data point $q_\phi(\mathrm{Z}) = \prod_{i=1}^{n} \mathcal{N}(\mu_i, \sigma_i)$. For the entire dataset (X, Z), the parameters of this distribution are determined by optimizing the Evidence Lower BOund (ELBO).

$$\begin{aligned}\log p_\theta(\mathrm{X}) &\geq \mathrm{E}_{q_\phi(\mathrm{Z})} \log \frac{p_\theta(\mathrm{X},\mathrm{Z})}{q_\phi(\mathrm{Z})} \\ &= \mathrm{E}_{q_\phi(\mathrm{Z})} \log p_\theta(\mathrm{X}|\mathrm{Z}) - D_{KL}(q_\phi(\mathrm{Z})||p_\theta(\mathrm{Z})), \end{aligned} \tag{2}$$

where the Kullback–Leibler divergence $D_{KL}(q||p) = \mathrm{E}_q \log \frac{q}{p}$ is a distance measure between two distributions q and p.

Traditional VI approximates a latent variable distribution $q_\phi(\mathrm{z}_i)$ for every data point i which becomes expensive or impossible when the number of data points n is large. VAEs circumvent this problem by learning a direct functional mapping from the data space to the latent space and vice versa. We will detail this method in the next section.

2.3.2. Variational Autoencoders

Instead of approximating n sets of parameters, VAEs employ so called inference networks to learn a functional mapping from the data space into the latent space. Thus, we define each latent variable to be determined by a distribution $\mathrm{z}_i \sim q_\phi(\mathrm{z}_i|\mathrm{x}_i)$ which is parameterized by a neural network (the inference network) that is a function of the data point x_i. In the Gaussian case this would imply that $\mathrm{z}_i \sim \mathcal{N}(\mu(\mathrm{x}_i), \sigma(\mathrm{x}_i))$, where $\mu(\cdot)$ and $\sigma(\cdot)$ are neural networks mapping from the data space to the parameter space of the latent variables. Likewise, the likelihood is represented by neural network mappings (the generative network) $\mathrm{x}_i \sim p_\theta(\mathrm{x}_i|\mathrm{z}_i)$. In this way, VAEs do not estimate n sets of parameters but only the parameters of the inference and generative network. These are optimized with help of the ELBO

$$\begin{aligned}\log p_\theta(\mathrm{X}) &\geq \mathcal{L}(\mathrm{X}, \theta, \phi) \\ &= \frac{1}{n}\sum_{i=1}^{n} \mathrm{E}_{q_\phi(\mathrm{z}_i|\mathrm{x}_i)} \log p_\theta(\mathrm{x}_i|\mathrm{z}_i) - D_{KL}(q_\phi(\mathrm{z}_i|\mathrm{x}_i)||p_\theta(\mathrm{z}_i)). \end{aligned} \tag{3}$$

Note that we replaced the expectation in Equation (2) with the Monte Carlo estimate summing over the individual data points.

3. METHODOLOGY

Following the introduction of VAEs above, we will now detail how we employ them to learn the sensorimotor patterns required for interactive tasks. We will begin with a description of human-human dynamics modeling which is subsequently used to guide the human-robot interaction model.

3.1. A Generative Model of Interaction

In general we assume that a recording *rec* consists of T_{rec} observations $\mathrm{x}^{s_1}_{1:T_{rec}}$ and $\mathrm{x}^{s_2}_{1:T_{rec}}$, where $(s_1, s_2) = (human_1, human_2)$, and x^s_t represents a single frame containing the joint positions of human $s \in \{s_1, s_2\}$. During testing time, we would like to be able to infer a future window (of size w) of observations after time t, i.e., we would like to predict $\mathrm{x}^{s_1}_{t:t+w}$ and $\mathrm{x}^{s_2}_{t:t+w}$. We assume a generative process that looks as follows

$$\begin{aligned} &\mathrm{x}^{s_1}_{t:t+w} \sim p_{\theta_x}(\mathrm{x}^{s_1}_{t:t+w}|\mathrm{z}^{s_1}_t), \quad \mathrm{z}^{s_1}_t \sim p_{\theta_z}(\mathrm{z}^{s_1}_t|\mathrm{d}_t), \\ &\mathrm{d}_t \sim p_{\theta_s}(\mathrm{d}_t|h^{s_1}_t), \quad h^{s_1}_t = f_\psi(h^{s_1}_{t-1}, \mathrm{x}^{s_1}_{t-1}) \end{aligned} \tag{4}$$

$$\begin{aligned} &\mathrm{x}^{s_2}_{t:t+w} \sim p_{\theta_x}(\mathrm{x}^{s_2}_{t:t+w}|\mathrm{z}^{s_2}_t), \quad \mathrm{z}^{s_2}_t \sim p_{\theta_z}(\mathrm{z}^{s_2}_t|\mathrm{d}_t), \\ &\mathrm{d}_t \sim p_{\theta_s}(\mathrm{d}_t|h^{s_2}_t), \quad h^{s_2}_t = f_\psi(h^{s_2}_{t-1}, \mathrm{x}^{s_2}_{t-1}). \end{aligned} \tag{5}$$

Here, the latent variables $\mathrm{z}^{s_1}_t$ and $\mathrm{z}^{s_2}_t$ for agent s_1 and s_2 encode the next time window $\mathrm{x}^{s_1}_{t:t+w}$ and $\mathrm{x}^{s_2}_{t:t+w}$, while $h^{s_2}_t$ is the deterministic output of a recurrent model f_ψ. The role of $h^{s_2}_t$ is to summarize the information contained in the past observations $t' < t$, which in turn is transformed into the shared task dynamics d_t. From a system perspective, d_t is the stochastic output of a neural network that driven by the hidden state vector $h^{s_2}_t$. As depicted in **Figure 2**, the d_t can be derived from the movement of either subject independently. These shared dynamics influence how each partner moves through $\mathrm{z}^{s_1}_t$ and $\mathrm{z}^{s_2}_t$. In summary, the generative model for agent s_1 represents the joint distribution $p_\theta(\mathrm{x}^{s_1}_{t:t+w}, \mathrm{z}^{s_1}_t, \mathrm{d}_t|h^{s_1}_t)$ conditioned on a deterministic summary of the past $h^{s_1}_t$ and parameterized by $\theta = (\theta_x, \theta_z, \theta_s)$.

In the following, we will describe how to learn each of the components for human-human interaction and subsequently how to transfer this knowledge to a human-robot interaction scenario.

3.1.1. Motion Embeddings

One problem when it comes to predicting the future is that there exist many possible ones. When using a mean-squared error based cost function during training, this will lead the model to rely on predicting only the average, not many different trajectories. We will circumvent this problem by first learning a latent space that encodes the future without knowledge of the past and then to learn a distribution over the latent variables which is conditioned on the past [e.g., $p(\mathrm{z}^{s_1}_t|\mathrm{d}_t)$ in Equation 5]. At each time step, we assume that there exists latent variables $\mathrm{z}^{s_1}_t$ and $\mathrm{z}^{s_2}_t$ for agent s_1 and s_2 which encode the next time window $\mathrm{x}^{s_1}_{t:t+w}$ and $\mathrm{x}^{s_2}_{t:t+w}$. We assume that both humans are encoded into a common space, therefore we will replace the superscripts s_1 and s_2 with s in the following discussion.

To infer the latent variables efficiently from data, we apply variational autoencoders (introduced in section 2.3). To this end, we define the following generative process:

$$\begin{aligned} &\mathrm{x}^s_{t:t+w} \sim p_{\theta_x}(\mathrm{x}^s_{t:t+w}|\mathrm{z}^s_t), \\ &\quad \mathrm{z}^s_t \sim p_{\theta_z}(\mathrm{z}^s_t) \\ &= \mathcal{N}(0,1), \text{ and approximate posterior } \mathrm{z}^s_t \sim q_{\phi_z}(\mathrm{z}^s_t|\mathrm{x}^s_{t:t+w}). \end{aligned} \tag{6}$$

The graphical model is depicted in **Figure 3A**. The parameters (θ_x, ϕ_z) of the generative network $p_{\theta_x}(\mathrm{x}^s_{t:t+w}|\mathrm{z}^s_t)$ and the inference network $q_{\phi_z}(\mathrm{z}^s_t|\mathrm{x}^s_{t:t+w})$ are jointly trained on the training data collected from both humans to optimize the Evidence Lower BOund (ELBO).

$$\begin{aligned} &\mathcal{L}(\mathrm{x}^s_{t:t+w}, \theta, \phi) = \mathbb{E}_{q_{\phi_z}(\mathrm{z}^s_t|\mathrm{x}^s_{t:t+w})} \log p_{\theta_x}(\mathrm{x}^s_{t:t+w}|\mathrm{z}^s_t) \\ &-D_{KL}(q_{\phi_z}(\mathrm{z}^s_t|\mathrm{x}^s_{t:t+w})||p_{\theta_x}(\mathrm{z}^s_t)). \end{aligned} \tag{7}$$

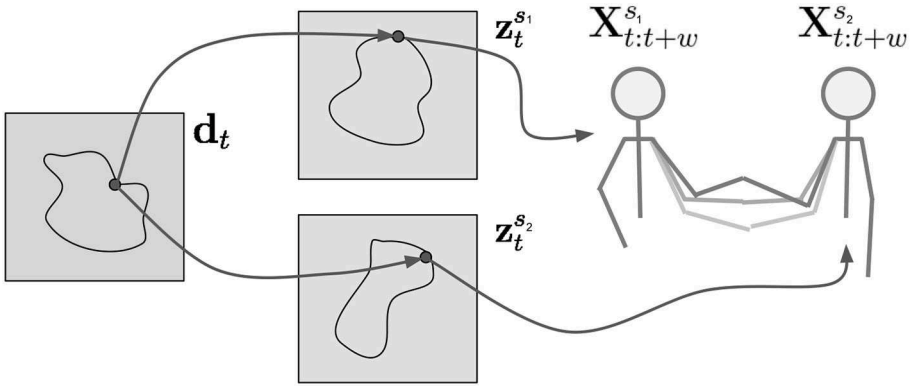

FIGURE 2 | The task dynamics d_t govern the activity of the latent variables of both partners $z_t^{s_1}$ and $z_t^{s_2}$. These in turn determine the future movement of the partners $x_{t:t+w}^{s_1}$ and $x_{t:t+w}^{s_2}$.

3.1.2. Encoding Task Dynamics

Once the motion embeddings have been learned, the whole generative model in Equation (5), as depicted in **Figure 3B**, can be trained. To this end, we need to infer the parameters $(\theta_z, \theta_s, \psi)$ to estimate $p_{\theta_z}(z_t^s|d_t)$, $p_{\theta_s}(d_t|h_t^s)$ and $f_\psi(h_{t-1}^s,\ x_{t-1}^s)$.

The loss function is defined as follows

$$\begin{aligned}&\mathcal{S}(x_{t-1:t+w}^{s_1}, x_{t-1:t+w}^{s_2}, \theta_z, \theta_s, \psi)\\&= \sum_{s\in\{s_1,s_2\}} D_{KL}(p_{\theta_z}(z_t^s|d_t)||q_{\phi_z}(z_t^s|x_{t:t+w}^s)) +\\&JSD(p_{\theta_s}(d_t|h_t^{s_1})||p_{\theta_s}(d_t|h_t^{s_2})).\end{aligned} \tag{8}$$

The first line in Equation (8) forces the distributions over latent variables z_t^s that depend on the past to be close to the expected motion embedding at time t. The second line enforces that the latent variable d_t, which encodes the task dynamics are the same for both agents. As the KL divergence is not symmetric, we use here the Jensen–Shannon divergence, which is defined as $JSD(p||q) = \frac{1}{2}(D_{KL}(p||\frac{1}{2}(p+q)) + D_{KL}(q||\frac{1}{2}(p+q)))$ for two distributions p and q.

3.1.3. Interactive Embodiment Mapping

Once trained, the generative model described above can be used to generate future trajectories for both agents given that only one agent has been observed. This is achieved by e.g., predicting the task dynamics variable $d_t \sim p_{\theta_s}(d_t|h_t^{s_1})$ with help of data collected for agent s_1 and using this variable to infer both $z_t^{s_1} \sim p_{\theta_z}(z_t^{s_1}|d_t)$ and $z_t^{s_2} \sim p_{\theta_z}(z_t^{s_2}|d_t)$. We will make use of this fact to infer not only a human partner's future movement, but also to guide how a robotic partner should react given the observed human.

As training data acquisition with a robot and a human in the loop is cumbersome and time consuming, we do not have access to as much training data of the human-robot interaction compared to the human-human interaction. Therefore, we will leverage the task dynamics representation learned from human-human interaction to guide the robot's corresponding motion commands. To this end, we extract the task dynamics distribution from the human partner for each time step of the human-robot interaction recordings and learn a mapping to the robot's motion commands with a second dynamics model.

In more detail, given a recording *rec* which consists of T_{rec} observations $x_{1:T_{rec}}^{s_1}$ and $x_{1:T_{rec}}^r$, where x_t^r represents the robot's state at time t, we first collect $d_{1:T_{rec}}$ which we set to the mean of the distribution $p_{\theta_s}(d_t|h_t^{s_1})$ for each time step t. We are now equipped with a training data set, containing the data point pairs $(x_{t:t+w}^r, d_t)$. In order to learn a predictive model from the task dynamic variable d_t to the future motion commands of the robot, $x_{t:t+w}^r$, we design a similar approach to the model described for human-human interaction. It includes a Variational Autoencoder functioning as a motion embedding and a recurrent network that encodes the robot motion over time. These two models are depicted in **Figures 3C,D**, respectively.

3.1.3.1. Interaction model with predictive input

Similar to the human-human setting in Equation (5), the generative model for the robot motion is as follows

$$\begin{aligned}x_{t:t+w}^r \sim\ p_{\theta_{xr}}(x_{t:t+w}^r|z_t^r), \quad z_t^r \sim p_{\theta_{zr}}(z_t^r|h_t^r),\\ h_t^r = f_{\psi_r}(h_{t-1}^r, x_{t-1}^r, d_{t-1}).\end{aligned} \tag{9}$$

Just as in the human-human setting, we first train a motion embedding VAE on the robot data, i.e., we train the following model with the same loss function as in Equation (7)

$$\begin{aligned}&x_{t:t+w}^r \sim\ p_{\theta_{xr}}(x_{t:t+w}^r|z_t^r), \quad z_t^r \sim p_{\theta_{zr}}(z_t^r)\\&= \mathcal{N}(0,1), \text{ and approx. posterior } z_t^r \sim q_{\phi_{zr}}(z_t^r|x_{t:t+w}^r).\end{aligned} \tag{10}$$

Subsequently, we assume that the parameters (θ_{zr}, ψ_r) in Equation (9) are inferred by optimizing

$$\mathcal{S}(x_{t-1:t+w}^{r_1}, d_t, \theta_z, \psi) = D_{KL}(p_{\theta_{zr}}(z_t^s|h_t^r)||q_{\phi_{zr}}(z_t^s|x_{t:t+w}^r)), \tag{11}$$

where the dynamics $d_t \sim p_{\theta_s}(d_t|h_t^{s_1})$ are extracted with help of the models trained on the human-human data. We summarize the training procedure of all our model in Algorithm 1.

Algorithm 1: All four steps of our combined motion embedding and dynamics modeling framework.

Human-human interaction

Data: $x^{s_1,s_2} = \{x^{s_1}_{1:T_{rec}}, x^{s_2}_{1:T_{rec}}\}_{rec \in \text{HHI recordings}}$

Step 1: Human motion embedding

Fit $p_{\theta_x}(x^s_{t:t+w}|z^s_t)$ and $q_{\phi_z}(z^s_t|x^s_{t:t+w})$ to x^{s_1,s_2}, following Equation 7.

Step 2: Task dynamics

Fit $p_{\theta_z}(z^s_t|\mathrm{d}_t), p_{\theta_s}(\mathrm{d}_t|h^s_t)$ and $f_\psi(h^s_{t-1}, x^s_{t-1})$ to x^{s_1,s_2}, following Equation 8.

Human-robot interaction

Data: $x^{s_1,r} = \{x^{s_1}_{1:T_{rec}}, \mathrm{d}_{1:T_{rec}}, x^r_{1:T_{rec}}\}_{rec \in \text{HRI recordings}}$, where $\mathrm{d}_t =$ mean of $p_{\theta_s}(\mathrm{d}_t|h^s_t)$

Step 3: Robot motion embedding

Fit $p_{\theta_{xr}}(x^r_{t:t+w}|z^r_t)$ and $q_{\phi_{zr}}(z^r_t|x^r_{t:t+w})$ to $x^{s_1,r}$, combining Equation 7 and 10.

Step 4: Interactive embodiment mapping

Fit $p_{\theta_{zr}}(z^r_t|h^r_t)$ and $f_{\psi_r}(h^r_{t-1}, x^r_{t-1}, \mathrm{d}_{t-1})$ to $x^{s_1,r}$, following Equation 11.

3.2. Generating Interactions

In order to employ our models during an ongoing interaction, we need to predict future time steps. As the dynamics and the motion embeddings encode a window of the next w time steps, the prediction up to this horizon is straight forward as it only requires a propagation of the observed data. To go beyond a time frame of w is made possible by our generative design. Instead of propagating observed data, one can let the models predict the next w time frames based on the observed data and provide these as an input to the model. In case of the human-robot interaction model, one has to first predict the human's future motion to extract the matching dynamics variables and can subsequently use these variables together with predictions of the robot's motion to generate long-term robot motion. During online interaction these predictions can be updated on the fly when new data is observed.

3.3. Baselines

We benchmark our approach on three baselines. Our own approach will be called *Human Motion Embedding* in the following.

The first baseline tests whether our predictive and adaptive approach is necessary or whether more static imitation learning techniques suffice. To test this, we group the robot trajectories in the training data according to action type and use Dynamic Time Warping (DTW) to align them. We fit Gaussian distributions with full covariance matrices to the trajectory of each of the robot's joints. If DTW resulted in a trajectory length of T_{DTW} for a certain action type and joint, then the Gaussian is of dimension T_{DTW}. A sample from each Gaussian model constitutes therefore a trajectory in joint angle space without input from the current human movement. We call this approach *Gaussian model*.

The second baseline tests whether our approach actually benefits from the encoded dynamics learned with the HHI data. Thus, in this case we train the same model as described in section 3.1.3. However, instead of feeding the dynamics variable d_t into the recurrent network $h^r_t = f_{\psi_r}(h^r_{t-1}, \mathrm{x}^r_{t-1}, \mathrm{d}_{t-1})$ in Equation (9), we feed the current human joint position x^s_{t-1}, i.e., $h^r_t = f_{\psi_r}(h^r_{t-1}, \mathrm{x}^r_{t-1}, \mathrm{x}^s_{t-1})$. This also affects the loss in Equation (11), which now is a function of x^s_{t-1}, i.e., $\mathcal{S}(\mathrm{x}^{r_1}_{t-1:t+w}, \mathrm{x}^s_{t-1}, \theta_{zr}, \psi_r)$. We call this approach *Raw Data HR* which symbolizes that we provide raw human and robot data as input to the model.

The third baseline tests whether the human observation is required at all or whether the approach is powerful enough to predict based on robot joint position alone. In this case we train the same model as described in section 3.1.3, but provide only the current robot joint positions x^r_{t-1}, i.e., $h^r_t = f_{\psi_r}(h^r_{t-1}, \mathrm{x}^r_{t-1})$. This also affects the loss in Equation (11), i.e., $\mathcal{S}(\mathrm{x}^{r_1}_{t-1:t+w}, \theta_{zr}, \psi_r)$. We call this approach *Raw Data R* which symbolizes that we provide only raw robot data as input to the model.

4. EXPERIMENTAL SETUP AND MODELS

In this section we describe the experimental setup as well as modeling decisions and the model training procedure. For more details regarding model architectures and model training, such as train and test splits (please see the **Supplementary Material**).

4.1. Task Description

Our interactive tasks consist of performing four different greeting gestures with a human partner. In each task execution we assume the identity of the gesture to be known apriori as the focus of this work lies on learning continuous interactive trajectories. However, our method can easily be extended to automatically infer the action type (Bütepage et al., 2018b). Two of the gestures fall into the category of dyadic leader–follower interaction, while the other two partners carry equal roles. The interactive gestures are detailed in **Table 1**. Between actions, the two partners are standing in an upright position with both arms directed downwards close to the body.

As the robot is not necessarily equipped with a hand-like gripper, the actions involving finger movement are omitted during human-robot interaction. Furthermore, we assume the robot to take the role of the follower.

4.2. Data Collection

We collected data from human-human and human-robot interaction, respectively. The robotic setup and the human motion recording setup are described below, followed by the data collection procedure.

4.2.1. Robotic System Setup

In this work, we use a dual-armed YuMi-IRB 14000 robot which has been developed by ABB specifically with human-robot collaboration in mind. As depicted in **Figure 4A**, each arm has seven joints Arm 1 (rotation motion), Arm 2 (bend motion), Arm 7 (rotation motion), Arm 3 (bend motion), Wrist 4 (rotation motion), Wrist 5 (bend motion), and Flange 6 (rotation motion). To control the robot, we work in the joint angle space, i.e., at each time step we have access to a seven dimensional vector consisting of radial measurements. To control the robot, we provide the system with the next expected joint angle configuration or a

TABLE 1 | Gesture descriptions for both equal and leader-follower roles.

Equal roles	
Hand waving:	Both: Lifting the right arm into an upright, 90-degree angle with the open palm facing the partner; moving the lower arm sideways in an oscillatory motion (around 3–6 cycles); lowering the arm.
Hand Shaking:	Both: Stretching the right arm forward to meet the partner's hand, grasping the partners hand; moving the lower arm up and down in an oscillatory motion (around 3–6 cycles); releasing the partner's hand, lowering the arm.
Leader-follower roles	
Parachute fist-bump:	Both: Stretching the right arm upwards with the hand closed to a fist to meet the partner's hand, touching the partner's fist with one's own; Leader: (parachute) Opening the hand and tilting it so that the flat, inner palm faces downwards; keeping the hand above the follower's hand; moving the hand in a slight sideways oscillatory motion while simultaneously moving downwards; Follower (person): Keeping the hand closed and slightly below the leader's hand; following the slight sideways oscillatory motion of the leader and moving the hand downwards; Both: Lowering the arm when the hand is approximately on the height of the hip.
Rocket fist-bump:	Both: Stretching the right arm downwards with the hand closed to a fist to meet the partner's hand, touching the partner's fist with one's own; Leader (rocket): Opening the hand slightly to point to fingers upwards; keeping the hand above the follower's hand; moving the hand upwards; Follower (fire): Opening the hand with all fingers stretched downwards; keeping the hand below the leader's hand; wiggling the fingers to simulate fire; moving the hand upwards; Both: Lowering the arm when the hand is approximately on the height of the shoulders.

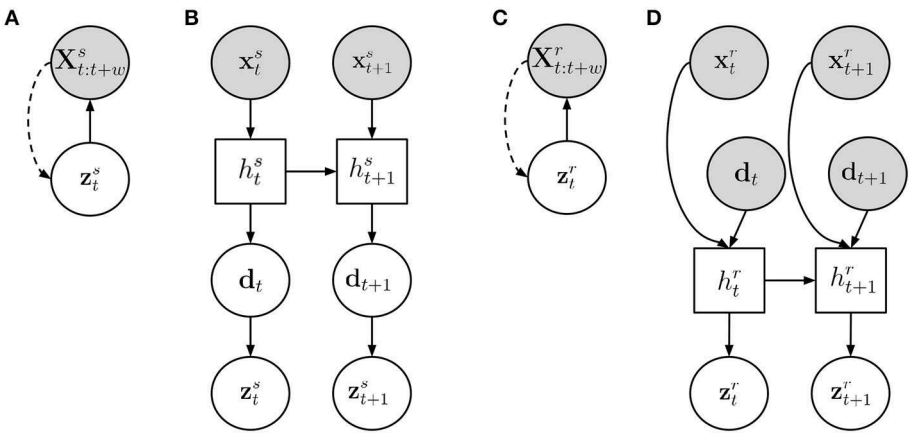

FIGURE 3 | An overview of the model structure. **(A)** Human motion embedding, **(B)** Task dynamics model, **(C)** Robot motion embedding, **(D)** Human motion embedding, Interaction model with predictive input. Gray circles represent observed variables, white circles are unobserved variables, and a white square indicates a deterministic unit. A filled line shows the generative process while dotted lines indicate inference connections.

FIGURE 4 | (A) The right arm of the Yumi robot used in the experiments. **(B)** A rokoko smart suit in action.

TABLE 2 | Statistics of the collected dataset.

	Human-human data			Human-robot data		
Action type	**# Trials**	**Min. duration (s)**	**Max. duration (s)**	**# Trials**	**Min. duration (s)**	**Max. duration (s)**
Hand shake	38	8.5	12.5	10	10.4	14.5
Hand wave	31	8.5	17.5	10	12.7	17.4
Parachute	49	7.0	12.0	11	11.0	14.3
Rocket	70	3.0	6.0	10	11.1	13.8

whole trajectory thereof. We sample the robot's joint angles at a frequency of 40 Hz.

4.2.2. Human Motion Capture

We recorded the 3D position of the human joints in Cartesian space during interaction with help of two Rokoko smart suits*. As shown in **Figure 4B**, these textile suits are equipped with 19 inertia sensors with which motion is recorded. Via wireless communication with a Wi-fi access point, the suits are able to record whole-body movements at a rate of up to 100 Hz. While simultaneous recordings with several suits are possible, we align the recordings offline. We record the 3D Cartesian positions of each joint in meters with respect to a body-centric reference frame. The data is sampled down to match the 40 Hz of the robot recording.

4.2.3. Collection Procedure

For the human-human dataset, we asked two participants to perform all four actions as described in section 4.1 for approximately 6 min each. The exact number of repetitions of each action type as well as duration statistics are listed in **Table 2**. A recording of the action *hand-shake* is depicted in the top of **Figure 5**.

For the robot-human dataset, we asked one of the participants to perform all four actions together with the robot. To this end, we made use of kinesthetic teaching, i.e., a human expert guided the arm of the robot during the interaction. As shown in **Table 2**, the duration of the human-robot trials is on average slightly longer than the human-human trials. A recording of the action *hand-shake* is depicted in the middle of **Figure 5**.

4.3. Modeling Decisions and Training Procedure

All models are implemented in Tensorflow (Abadi et al., 2015). Instead of training four separate models, one for each action, we train a single model that can generate all actions. In order to signal to the model, which action is currently performed, we encode the actions as a one-hot vector which is passed as an additional input to the model as described below.

*https://www.rokoko.com/

4.3.1. Modeling Choices

All latent variables $(z_t^{s1}, z_t^{s2}, z_t^r, d_t^s)$ are chosen to be independent and identically distributed Gaussian units with a trainable mean and variance. The prior of the VAEs is set to be standard normal distributed $p_{\theta_z}(z_t) \sim \mathcal{N}(0, 1)$.

To indicate to the recurrent models which action is currently performed, we provide the networks with a one-hot vector indicating the current action. We add an additional *not-active* action, which indicates those time steps after completion of the interaction. Thus, the one-hot vector is of dimension 5 and is concatenated with the observed joint positions of either human or robot.

We train two identical models for the two human partners while the model of the robot motion has a different structure. Please see the **Supplementary Material** for details about model architecture.

4.3.2. Data Representation

We represent the human by four joints "RightShoulder," "RightArm," "RightForeArm," and "RightHand" in 3D Cartesian space, resulting a 12 dimensional vector. We center the arm around the shoulder joint. The robot is represented by a seven dimensional vector, each indicating a joint angle. We select 80% of all trials of a certain interaction as training data and keep 20% as testing data. In practice, we keep the last 20% of trials of the recording. This results in 149 trials as training data and 39 trials of testing data for the HHI recordings in 32 trials as training data and 9 trials of testing data for the HRI recordings.

4.3.3. Training Details

For optimization we use the Adagrad optimizer with a learning rate of 0.001. The batch size is 5,000 for the VAEs and 500 for the recurrent networks. If a dataset does not contain that many samples, we replicate the training samples to get to 5,000. We train all models until convergence. For the VAEs we use a form of β-VAE (Higgins et al., 2017), where $\beta = 0.5$. For training the recurrent networks, we pad all data sequences with ones to have the same length.

5. RESULTS

In this section we present the performance of the proposed approach. Online employment of our approach during the action *hand-shake* is depicted in the bottom of **Figure 5**. More examples can be found in the Supplementary Material in form of a video **(Supplementary Video 1)**. In the analysis we present results on held-out test datasets. As described in section 4.3.1, each model was trained on all actions simultaneously and subsequently tested on each of the actions in the held-out test dataset.

We begin by investigating the predictive performance of the models trained on the human-human dataset. This will be followed by an analysis of the robot motion prediction. In this case, we consider both the predictive error as well as the entrainment of predicted vs. ground truth robot motion with the human motion.

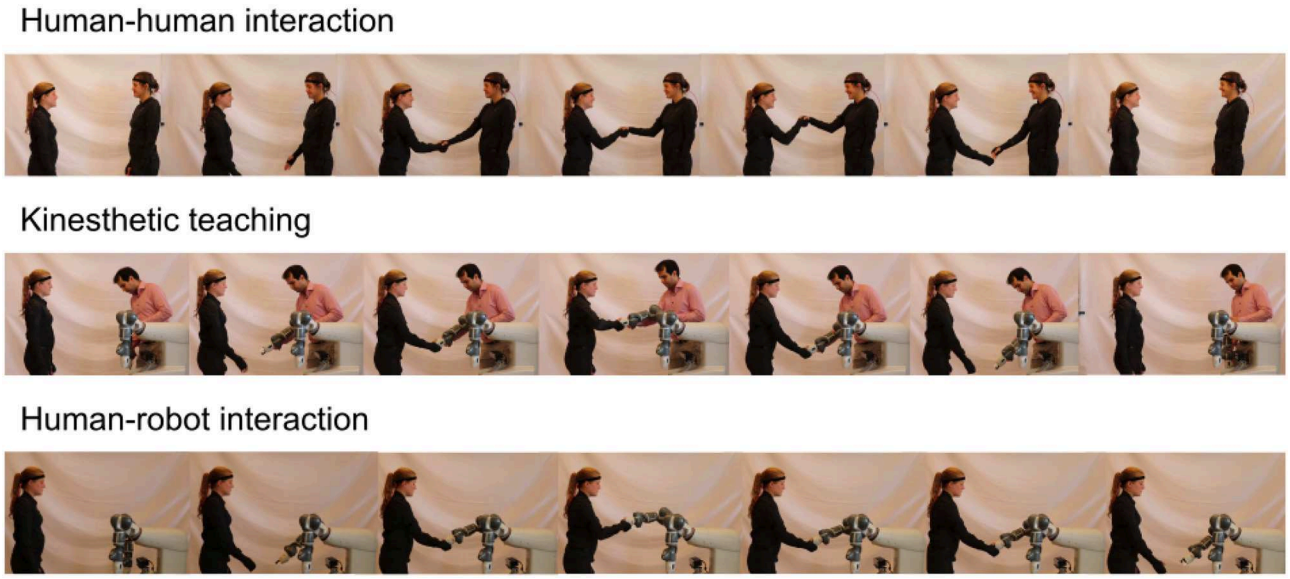

FIGURE 5 | The data collected during human-human interaction **(top)** and kinesthetic teaching **(middle)** is used train the proposed models. These are employed in human-robot interaction tasks **(bottom)**.

5.1. Predictive Performance on Human-Human Data

We have two reasons for collecting additional human-human interaction data. Firstly, we hypothesize that the dynamics learned based on HHI data can guide robot action selection during HRI experiments. Secondly, it is easier to collect HHI data, allowing for larger datasets. To test the second hypothesis we trained the human motion embedding and dynamics models both on HHI data and only on the human data contained in the HRI data. In the latter case, the dynamics variable is not restricted to match a human partner. We test the predictive capacity of both these models by computing the mean squared prediction error (MSPE) for the time window w on both test data sets (HRI and HHI). The results are depicted in **Figure 6**. Two observations can be made. First of all, the model trained on HRI data does not generalize well, mainly caused by the small training data set. Secondly, the prediction error does not increase drastically over time as should be expected. Due to the fact that we do not force the model to predict a whole trajectory as e.g., (Bütepage et al., 2018a) but only a latent variable which can be decoded into a trajectory, our model is less prone to regress to the mean but to encode the actual motion.

5.2. Predictive Performance on Human-Robot Data

In this section we inspect how our proposed dynamics transfer approach performs against the baselines. As the different joints move to different extents, the range of joint angles varies. Therefore, we measure the predictive error not with the MSPE as in the case of HHI predictions but with the normalized root-mean-square deviation (NRMSD) which is computed as follows:

TABLE 3 | NRMSD computed on robot testing data averaged over all joints.

Human motion embedding	Raw data RH	Raw data R	Gaussian model
NRMSD computed on robot testing data			
0.16	0.22	0.18	0.20

The bold value indicates which model performed best.

$$NRMSD(\{\mathrm{x}^r_{1:T_{tr},j}\}_{tr\in 1:TR}, \{\hat{\mathrm{x}}^r_{1:T_{tr},j}\}_{tr\in 1:TR}) = \frac{1}{TR}\sum_{tr\in 1:TR}\sqrt{\frac{1}{T_{tr}(j_{max}-j_{min})}\sum_{t=1}^{T_{tr}}(\mathrm{x}^r_{t,j}-\hat{\mathrm{x}}^r_{t,j})^2}, \quad (12)$$

for the *jth* joint. Here *tr* denotes trial, i.e., one execution of an interaction, and TR is the number of trials. j_{max} and j_{min} denote the maximum and minimum value that has been recorded for the *jth* joint in the training data.

We start by comparing our approach (Human Motion Embedding) to the two models that have an identical structure but that differ in the type of input data (Raw Data HR and Raw Data R). To this end, we provide ten time steps as input to the models and let the recurrent network predict 30 steps as described in section 3.2. This process is repeated until the end of a trial is reached. Since the Raw Data HR model is not able to generate human motion, we provide it with the last observed human pose. Through the motion embedding, the models produce a prediction of the next 40 time steps (1 s). We average over all time steps and present the results in **Figure 7**. The Human Motion Embedding appears to produce the smallest errors, especially for those joints that are vital for the interaction

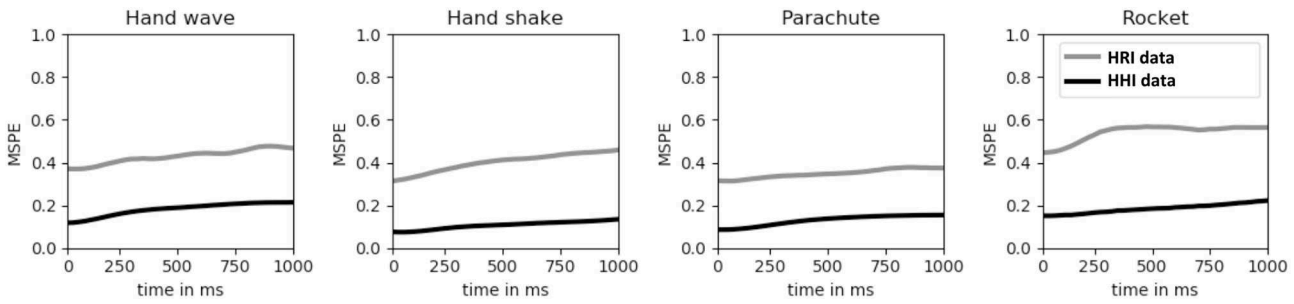

FIGURE 6 | The mean squared prediction error (MSPE) in meters for human-human interaction over a time horizon of 1 s. The human motion embedding and dynamics models were trained on the HHI data and on the human data contained in the HRI data set. In the latter case, the dynamics variable is not restricted to match a human partner.

(joint 2, 3, and 4). The wrist joints (joint 6 and 7) are of less importance and do also show a larger degree of between-trial variance in the training data. We depict the predictions of each of the Human Motion Embedding model, the Raw Data HR model and the ground truth trajectory for one testing trial of each action in **Figure 8**.

When averaged over the forty time steps of prediction, the difference becomes clear in **Table 3**, where we also include the Gaussian model. As the Raw Data HR model is not able to predict human motion, it produced the largest error. The Human Motion Embedding outperforms both the adaptive Raw Data HR and Raw Data R models as well as the non-adaptive Gaussian model. The adaptive Raw Data R model produces a smaller error than the non-adaptive Gaussian model, which also is trained on raw robot data. We will investigate the difference between adaptive and non-adaptive approaches in more detail in the next section.

5.3. Non-adaptive vs. Adaptive Motion Generation

As discussed in section 2.2, Human-Robot interaction has additional requirements compared to traditional imitation learning. It does not suffice to learn a distribution over the trajectories observed in the training data and sample a whole trajectory during run-time. Instead, HRI requires adaptive and predictive models that react to the human's actions such that a sensorimotor coupling between human and robot can arise. We visualize this in **Figure 9** by sampling from the Gaussian model of joint 4 for the action hand-shake. It becomes apparent that none of the samples is in accordance with any of the testing trials that are also depicted. First of all, the motion onset differs and the duration of the trajectory is predetermined due to the time alignment, while the duration of natural interaction differs from trial to trial. Additionally, the movement is not adapted to the human's hand-shake but has different degrees of phase shift. If we compare these predictions to the predictions of joint 4 in the second row of **Figure 8**, we realize that the adaptive approach reacts in a timely manner and follows the oscillations of the ground truth motion that match the human motion. We will investigate the degree of entrainment of the predictions of robot with the human motion in the next section.

5.4. Entrainment on Human-Robot Data

With this work we are aiming at developing models that allow for sensorimotor coupling between humans and robots to benefit physical HRI. We visualize the generated predictions of the Human Motion Embedding model as well as the ground truth robot motion data and the hand position of the human for a testing trial of each interaction in **Figure 10**. As not all joints are relevant to a task, we visualize joint 2–4. We see that the predicted motion follows the oscillatory movement of the human hand during hand-wave (see joint 3), hand-shake (see joint 4), and the parachute (see joint 4) interaction as well the rise and fall of the rocket action (see joint 2 and 3).

To investigate whether the models capture this coupling, we extract the dynamics variables of the human motion of an entire testing trial of the hand-shake interaction as well as the latent variables that predict the robot motion. We then apply factor analysis to these two streams of data and compare the two first components to each other. The two components are visualized in **Figure 11**. The first factor appears to represent the general onset, duration and offset of the interaction while the second factor encodes the oscillatory motion of the hand and arm. We see that, although the factor analysis is performed independently on the human and robot latent variables, the overall structure is similar. Additionally, the oscillatory motion is overlapping, indicating a coupling between the two systems.

6. CONCLUSION

In this work, we propose a deep generative model approach to imitation learning of interactive tasks. Our contribution is a novel probabilistic latent variable model which does not predict in joint space but in latent space, which minimizes the chance of regression to the mean. We employ this model both as a dynamics extractor of HHI as well as the basis for the motion generation of a robotic partner. Our experiments indicate that HRI requires adaptive models which take the human motion and task dynamics into account. These dynamics, which encode the movement of both humans (see **Figure 2**), and therefore the

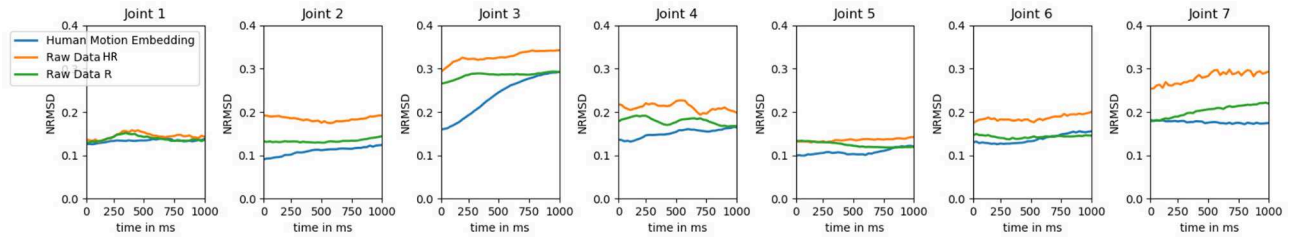

FIGURE 7 | The normalized root-mean-square deviation (NRMSD) for robot motion during human-robot interaction over a time horizon of 1 s. The error is averaged over the time steps of all trials. The models are provided with 10 time steps as input and the recurrent network predict 30 steps as described in section 3.2. This process is repeated until the end of a trial is reached. The Raw Data HR model receives the last observed human pose as it is not able to generate human motion.

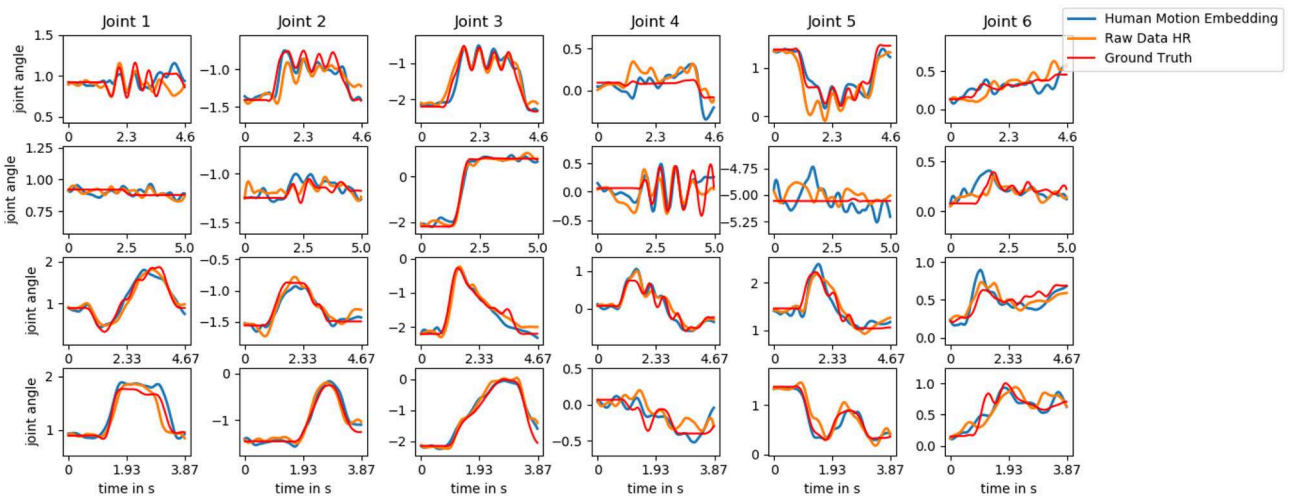

FIGURE 8 | The joint angle trajectory of joint 1–6 for a testing trial of each of the actions *hand-wave, hand-shake, parachute*, and *rocket* (top to bottom). We depict the ground truth data (red) compared to the predicted trajectory of the Human Motion Embedding model (blue) and the Raw Data HR model (orange). The predictions are performed as described in section 3.2 after initializing the models with ten observation steps. The Raw Data HR model is not able to generate human motion trajectories and is therefore provided with the last observed human pose.

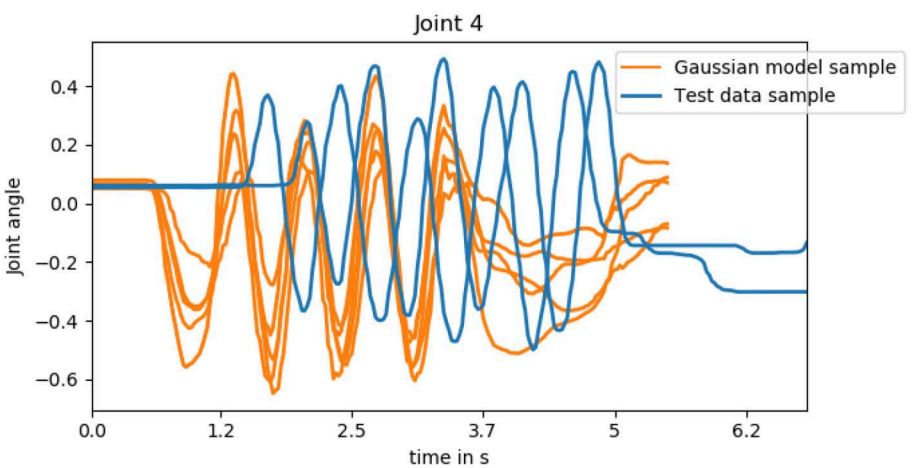

FIGURE 9 | Five samples from the Gaussian model of joint 4 of the hand-shake action and two testing data trials. As the Gaussian model is not adaptive, none of the samples is in accordance with any of the testing trials that are also depicted which is apparent in both the onset of motion and the phase shift.

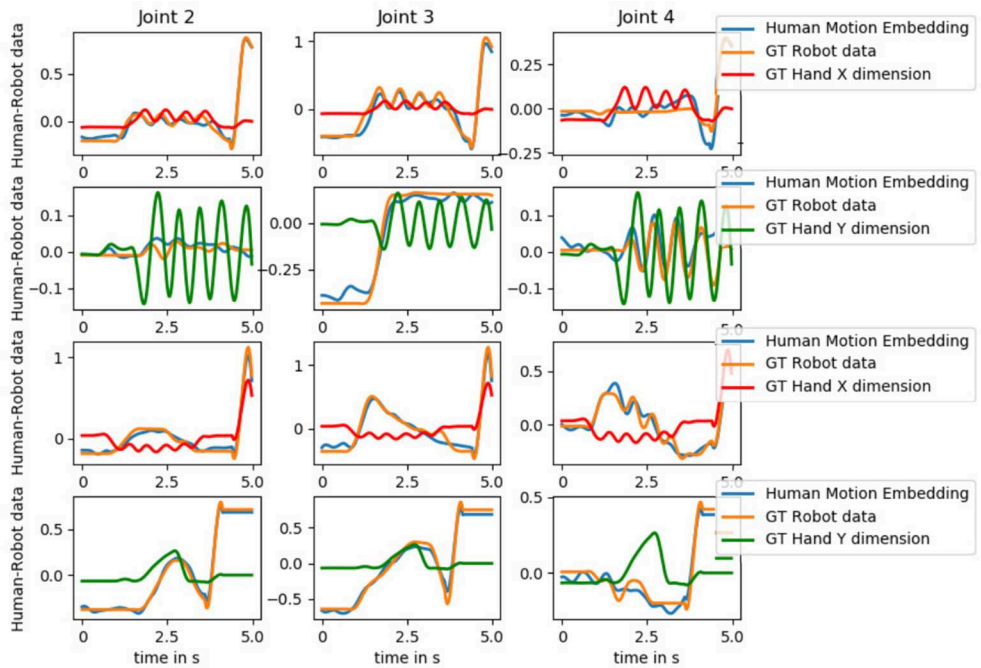

FIGURE 10 | The predictions of the Human Motion Embedding model as well as the ground truth robot motion data and the hand position (X or Y dimension) of the human for a testing trial of each interaction *hand-wave, hand-shake, parachute*, and *rocket* (top to bottom). The values are normalized to facilitate comparison. As not all joints are relevant to a task, we visualize joint 2–4. The predicted motion follows the oscillatory movement of the human hand during hand-wave (see joint 3), hand-shake (see joint 4), and the parachute (see joint 4) interaction as well the rise and fall of the rocket action (see joint 2 and 3).

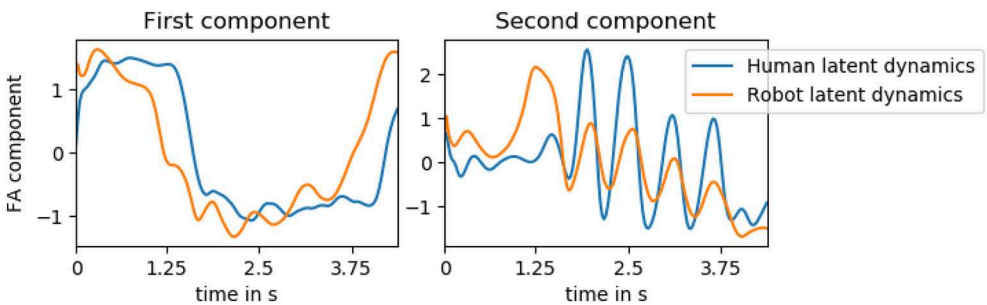

FIGURE 11 | The first two factor analysis (FA) components of a testing trial of the hand-shake interaction computed on both the latent variables extracted from the human ground truth motion and the latent variables predicting the robot motion. The first factor appears to represent the general onset, duration, and offset of the interaction while the second factor encodes the oscillatory motion of the hand and arm. Although the factor analysis is performed independently on the human and robot latent variables, the overall structure is similar. Additionally, the oscillatory motion is overlapping, indicating a coupling between the two systems.

coupling of the human partners during interaction, guide the generation of the robot which thus is coupled to its human partner.

After having established that the cheaper HHI data is required for high predictive performance (see section 5.1), we demonstrate that the extracted dynamics facilitate the performance of the predictive model of robot motion (see section 5.2). This indicates that the encoding of the future human motion and task dynamics can contribute to the robot's motion planning. This is in contrast to common approaches to imitation learning for interaction which use non-adaptive models. As we discuss in section 5.3, a non-adaptive trajectory model does not suffice in interactive tasks such as hand-shaking. With help of our generative approach, we can create synchronized behavior which shows a level of entrainment between human and robot (see section 5.4).

We believe that prediction and adaptation are essential to allow for natural HRI in shared workspaces. In future work, we plan to employ the system in real-time and to extend it to more complex tasks.

DATA AVAILABILITY STATEMENT

The datasets for this study can be found in the transfer learning for interaction repository, https://github.com/jbutepage/human_robot_interaction_data.

ETHICS STATEMENT

Ethical review and approval was not required for the study on human participants in accordance with the local legislation and institutional requirements. The patients/participants provided their written informed consent to participate in this study. Written informed consent was obtained from the individual(s) for the publication of any potentially identifiable images or data included in this article.

AUTHOR CONTRIBUTIONS

JB contributed to the idea development and data collection, developed the methodology, implemented and trained the models, evaluated the models, and wrote the manuscript. AG contributed to the idea, data collection, and development of the robot software. ÖÖ contributed to the data collection and implementation of other baselines. MB and DK supervised the work.

FUNDING

This work was supported by the EU through the project socSMCs (H2020-FETPROACT-2014) and the Swedish Foundation for Strategic Research and EnTimeMent (H2020-FETPROACT-824160), and the Knut and Alice Wallenberg Foundation.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/frobt.2020.00047/full#supplementary-material

Supplementary Video 1 | The video demonstrates data collection for human-human and human-robot interaction as well as online employement of the trained models.

REFERENCES

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., et al. (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Available online at: tensorflow.org.

Alissandrakis, A., Nehaniv, C. L., and Dautenhahn, K. (2007). Correspondence mapping induced state and action metrics for robotic imitation. *IEEE Trans. Syst. Man Cybernet. Part B* 37, 299–307. doi: 10.1109/TSMCB.2006.886947

Argall, B. D., Chernova, S., Veloso, M., and Browning, B. (2009). A survey of robot learning from demonstration. *Robot. Auton. Syst.* 57, 469–483. doi: 10.1016/j.robot.2008.10.024

Billard, A., Calinon, S., Dillmann, R., and Schaal, S. (2008). "Robot programming by demonstration," in *Springer Handbook of Robotics*, eds B. Siciliano and O. Khatib (Secaucus, NJ: Springer), 1371–1394.

Brownell, C. A. (2011). Early developments in joint action. *Rev. Philos. Psychol.* 2, 193–211. doi: 10.1007/s13164-011-0056-1

Bütepage, J., Kjellström, H., and Kragic, D. (2018a). "Anticipating many futures: online human motion prediction and generation for human-robot interaction," in *2018 IEEE International Conference on Robotics and Automation (ICRA)* (Brisbane, QLD: IEEE), 1–9. doi: 10.1109/ICRA.2018.8460651

Bütepage, J., Kjellström, H., and Kragic, D. (2018b). Classify, predict, detect, anticipate and synthesize: hierarchical recurrent latent variable models for human activity modeling. *arXiv[Preprint]. arXiv:1809.08875*.

Calinon, S., D'halluin, F., Sauser, E. L., Caldwell, D. G., and Billard, A. G. (2010). Learning and reproduction of gestures by imitation. *IEEE Robot. Autom. Mag.* 17, 44–54. doi: 10.1109/MRA.2010.936947

Dautenhahn, K. (2007). Socially intelligent robots: dimensions of human–robot interaction. *Philos. Trans. R. Soc. B Biol. Sci.* 362, 679–704. doi: 10.1098/rstb.2006.2004

Dinh, L., Sohl-Dickstein, J., and Bengio, S. (2017). "Density estimation using real NVP," in *International Conference on Learning Representations (ICLR)* (Toulon).

Dong, S., and Williams, B. (2011). Motion learning in variable environments using probabilistic flow tubes. *Int. J. Soc. Robot.* 4, 357–368. doi: 10.1109/ICRA.2011.5980530

Dong, S., and Williams, B. (2012). Learning and recognition of hybrid manipulation motions in variable environments using probabilistic flow tubes. *Int. J. Soc. Robot.* 4, 357–368. doi: 10.1007/s12369-012-0155-x

Ghadirzadeh, A., Maki, A., Kragic, D., and Björkman, M. (2017). "Deep predictive policy training using reinforcement learning," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Vancouver, BC: IEEE), 2351–2358. doi: 10.1109/IROS.2017.8206046

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., et al. (2014). "Generative adversarial nets," in *NIPS* (Montreal, QC).

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., et al. (2017). "beta-VAE: learning basic visual concepts with a constrained variational framework," in *International Conference on Learning Representations (ICLR)* (Toulon), 6.

Kingma, D. P., and Dhariwal, P. (2018). "Glow: generative flow with invertible 1x1 convolutions," in *Advances in Neural Information Processing Systems* (Montreal, QC), 10236–10245.

Kingma, D. P., and Welling, M. (2015). "Auto-encoding variational bayes," in *International Conference on Learning Representations (ICLR)* (San Diego, CA).

Koppula, H. S., and Saxena, A. (2015). Anticipating human activities using object affordances for reactive robotic response. *IEEE Trans. Pattern Anal. Mach. Intell.* 38, 14–29. doi: 10.1109/TPAMI.2015.2430335

Li, Y., Song, J., and Ermon, S. (2017). "Infogail: Interpretable imitation learning from visual demonstrations," in *Advances in Neural Information Processing Systems* (Montreal, QC), 3812–3822.

Maeda, G., Ewerton, M., Neumann, G., Lioutikov, R., and Peters, J. (2017a). Phase estimation for fast action recognition and trajectory generation in human–robot collaboration. *Int. J. Robot. Res.* 36, 1579–1594. doi: 10.1177/0278364917693927

Maeda, G. J., Neumann, G., Ewerton, M., Lioutikov, R., Kroemer, O., and Peters, J. (2017b). Probabilistic movement primitives for coordination of multiple human–robot collaborative tasks. *Auton. Robots* 41, 593–612. doi: 10.1007/s10514-016-9556-2

Osa, T., Pajarinen, J., Neumann, G., Bagnell, J. A., Abbeel, P., Peters, J., et al. (2018). An algorithmic perspective on imitation learning. *Found. Trends® Robot.* 7, 1–179. doi: 10.1561/2300000053

Rezende, D. J., Mohamed, S., and Wierstra, D. (2014). "Stochastic backpropagation and approximate inference in deep generative models," in *International Conference on Machine Learning* (Beijing), 1278–1286.

Rogoff, B., Mistry, J., Göncü, A., Mosier, C., Chavajay, P., and Heath, S. B. (1993). Guided participation in cultural activity by toddlers and caregivers. *Monogr. Soc. Res. Child Dev.* 58, v–vi, 1–174; discussion: 175–179. doi: 10.2307/1166109

Sakoe, H., and Chiba, S. (1978). Dynamic programming algorithm optimization for spoken word recognition. *IEEE Trans. Acoust. Speech Signal Proc.* 26, 43–49. doi: 10.1109/TASSP.1978.1163055

Sebanz, N., Bekkering, H., and Knoblich, G. (2006). Joint action: bodies and minds moving together. *Trends Cogn. Sci.* 10, 70–76. doi: 10.1016/j.tics.2005.12.009

Vesper, C., Abramova, E., Bütepage, J., Ciardo, F., Crossey, B., Effenberg, A., et al. (2017). Joint action: mental representations, shared information and general mechanisms for coordinating with others. *Front. Psychol.* 7:2039. doi: 10.3389/fpsyg.2016.02039

Zhang, C., Butepage, J., Kjellstrom, H., and Mandt, S. (2018). Advances in variational inference. *IEEE Trans. Pattern Anal. Mach. Intell.* 41, 2008–2026. doi: 10.1109/TPAMI.2018.2889774

Zhang, T., McCarthy, Z., Jow, O., Lee, D., Chen, X., Goldberg, K., et al. (2018). "Deep imitation learning for complex manipulation tasks from virtual reality teleoperation," in *2018 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE), 1–8. doi: 10.1109/ICRA.2018.8461249

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.