A System for Learning Continuous Human-Robot Interactions from Human-Human Demonstrations

Conference Paper · May 2017

DOI: 10.1109/ICRA.2017.7989334

5 authors, including:

David Vogt
LogMeIn Dresden
26 PUBLICATIONS **472** CITATIONS
SEE PROFILE

Simon Stepputtis
Arizona State University
14 PUBLICATIONS **106** CITATIONS
SEE PROFILE

Steve Grehl
Technische Universität Bergakademie Freiberg
22 PUBLICATIONS **158** CITATIONS
SEE PROFILE

A System for Learning Continuous Human-Robot Interactions from Human-Human Demonstrations

David Vogt[1], Simon Stepputtis[2], Steve Grehl[1], Bernhard Jung[1], Heni Ben Amor[2]

***Abstract*— We present a data-driven imitation learning system for learning human-robot interactions from human-human demonstrations. During training, the movements of two interaction partners are recorded through motion capture and an interaction model is learned. At runtime, the interaction model is used to continuously adapt the robot's motion, both spatially and temporally, to the movements of the human interaction partner. We show the effectiveness of the approach on complex, sequential tasks by presenting two applications involving collaborative human-robot assembly. Experiments with varied object hand-over positions and task execution speeds confirm the capabilities for spatio-temporal adaption of the demonstrated behavior to the current situation.**

I. Introduction

Robot co-workers that work alongside human partners are a major vision of robotics and artificial intelligence. Robotic assistants with these capabilities could radically transform today's workplaces in manufacturing, healthcare, and the services sector. However, for this vision to become reality, an algorithmic foundation is needed, which allows for the specification of collaborative interactions between humans and robots. In the traditional robot programming paradigm, a human engineer would be required to foresee all important interaction parameters and implement control routines that generate appropriate robot responses. Unfortunately, even for moderately complex interaction scenarios this approach becomes intractable. This limitation is particularly true for physical and continuous interaction scenarios that are not based on turn-taking between the partners, e.g., joint transportation or collaborative manipulation of an object.

In this paper, we propose to extract the interaction dynamics for a collaborative task through *learning by demonstration*. Our objective is to develop methodologies, that allow robots to gradually increase their repertoire of interaction skills without additional effort for a human programmer. Whereas current imitation learning methods almost exclusively focus on a single agent, our method is based on parallel behavior demonstrations by two interaction partners. In particular, the proposed approach inherently captures the important spatial relationships and synchrony of body movements between two interacting partners.

Previous work has shown that Interaction Meshes (IM) [1] are well suited for the spatial adaption of interaction examples in virtual agent animation and interactive human-robot settings [1]–[4]. Similarly, it has been shown that the temporal relationships during interactions can be efficiently extracted using hidden Markov models (HMM) [5]–[7]. However, recent results also indicate that models based on HMMs alone do not generalize sufficiently to postural changes in typical human-robot interaction tasks [8]. As a result, further optimizations to the robot posture and movement are required to ensure efficient and safe physical interaction.

Fig. 1. In a collaborative assembly task, the robotic assistant continuously coordinates its behavior with the human co-worker. The robot's behavior is learned from human-human demonstrations of the various subtasks.

In this paper, we present a system that allows for adequate spatio-temporal adaptation of recorded human-human interactions during HRI scenarios. The approach builds upon previous results [1], [4], [8] and extends them to complex physical interaction scenarios, e.g., collaborative assembly. In particular, we show that complex spatio-temporal generalization can be achieved through the combination of motion recognition in low-dimensional posture spaces and a variant of IMs for real-time robot response generation.

Our contributions can be summarized as follows:

- A methodology for extracting human-robot interaction from human-human demonstrations
- Correlation-based Interaction Mesh generation in HRI settings
- Temporally adaptive motion recognition in low-dimensional posture spaces using a probabilistic method and local alignment.

The proposed approach is evaluated w.r.t. its capabilities for spatial and temporal adaptation in two complex assembly tasks involving human-robot collaboration.

[1]David Vogt, Steve Grehl and Bernhard Jung are with Faculty of Mathematics and Informatics, Technical University Bergakademie Freiberg, 09599, Freiberg, Germany surname.name@informatik.tu-freiberg.de

[2]Simon Stepputtis and Heni Ben Amor are with the School of Computing, Informatics, Decision Systems Engineering, Arizona State University, 660 S. Mill Ave, Tempe, AZ 85281 sstepput/hbenamor@asu.edu

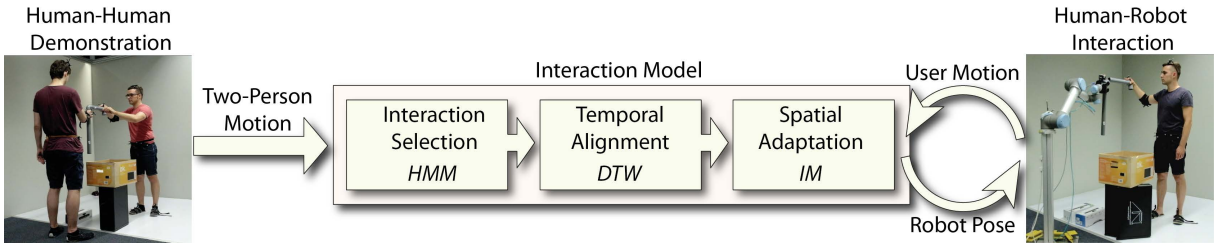

Fig. 2. Overview of the proposed method. Based on human-human demonstrations we learn how each person moved during the joint execution of a physical collaboration in an interaction model. Using the learned model in a human-robot interaction we infer the robot's role and compute its controls continuously. The broad context of the interaction is first derived temporally and then aligned locally in low-dimensional space. Spatial adaptation based on matched interaction templates is achieved using IMs.

II. Related Work

Bootstrapping the behavior repertoire of robots through demonstration has been a prominent approach in robotics, with a large body of papers dedicated to the topic [9]–[11]. Recent research instead focuses on transferring this paradigm to collaborative scenarios in which robots engage in cooperative tasks with human partners. To that end, however, correlations and interdependencies among the interaction partners need to be carefully extracted and imitated.

Rozo and colleagues propose a time-dependent approach for mimicking observed forces and positions during object transportation tasks [12]. A robotic arm is kinesthetically trained to match a user's hand position while carrying an object. A Gaussian mixture model (GMM) is computed to approximate task-specific parameters. The model is later employed during runtime to infer velocities, positions and forces in ongoing interactions. However, when doing so, the user is limited to the trained interaction speed. Our framework on the other hand generalizes demonstrated tasks and, thus, allows spatiotemporal variations thereof.

Based on the general concept of dynamic movement primitives (DMP), Ben Amor and colleagues propose a *mixture of interaction primitives*, a compact representation of human-human demonstrations [13]. By maintaining a distribution over DMP parameters, inherent correlations of the joint task are encoded. In doing so, a robotic arm learns to react to human motions during joint task execution.

Similar to our work, direct marker control is also being used in [14]. Recorded human motions are adapted by attaching virtual springs to the robots' extremities which in turn pull joints towards recorded marker positions. During physical human-robot interaction an additional virtual spring is placed between the leading hand of the user as well as the robot. However, due to the single connection between both interactants only a limited amount spatial generalization is accounted for. Compared with this, the proposed system provides a more generic strategy to spatial adaptation and includes complex inter-person connections.

An alternative approach for spatial relationship preservation between interactants are IMs [1]. First introduced within the computer animation community to animate virtual characters in realtime [15], [16], IMs are also applied in the field of robotics. Ivan and colleagues, for example, compute optimal motion paths for a robotic arm in semi-dynamic environments [2]. However, changes in the environment are only incorporated offline, thus rendering the approach not suitable for real-time human-robot interactions.

In contrast to above approaches, we propose an approach that combines the strength of IMs in preserving the spatial relationships between multiple body parts with a multi-stage approach for real-time motion recognition and temporal adaption of robot movements.

III. Methodology

We first record *human-human* demonstrations of two users performing cooperative tasks using motion capture. Several tasks can be demonstrated and only one example demonstration per task is necessary. Generally, we assume leader-follower type scenarios, where one person acts as an assistant. During the later human-robot interaction, the robot will assume the role of the assistant.

In the training phase, an interaction model is learned that describes how the two interactants synchronize their movements. At runtime, the interaction model is used to continuously adapt the robot's movements to the human interactant.

For notational clarity, following [8] we will refer to the first interaction partner, i.e., the human, as the *observed agent*, while the second interaction partner, i.e., the robot, will be called *controlled agent*.

IV. The Interaction Model

The interaction model presented in this section serves to identify a controlled agents response to the movement of the observed agent in cooperative tasks. From a methodological point of view, the model provides means of interaction selection and spatio-temporal adaptation so that demonstrated tasks can be optimized to unknown human-robot interactions (see Fig. 2). Structurally, the interaction model consists of a large database of Interaction Meshes (IMs) as well as data structures for identifying the best matching IM during an ongoing human-robot interaction. Each IM represents a pair of postures of the human-human demonstration at a single time step and it captures the spatial relationships of the interaction. We developed a correlation-based variant of IMs that focus on the most relevant joints of the two human demonstrators in order to increase postural generalization.

During model learning a *global posture space*, in conjunction with an HMM is computed based on a single task demonstration. Then, for each demonstration a low-dimensional *local posture space* is defined and further segmented into smaller parts of the interaction before finally an IM is constructed for each pair of poses.

During runtime, a suitable human-human interaction is selected based on observed user poses and the learned HMM. Then, the user's motion is temporally aligned in the local posture space of the inferred interaction demonstration. Utilizing the context-dependent IM that is associated with the matching time frame, a robot's pose is finally optimized to best-fit the current situation.

A. Local Posture Spaces and their Segmentation

For each interaction i, a low-dimensional local posture space $\mathcal{L}_i$ is calculated using principal component analysis (PCA) based on the observed agent's motion capture data. The observed agent's motion is, thus, compactly represented as a trajectory in the low-dimensional space $\mathcal{L}_i$. The trajectories in $\mathcal{L}_i$ are then further segmented using Hotellings T-squared statistics. A segment $\mathcal{Q}_m$ with $m \in \{1, \ldots, M\}$ is defined as sequence of consecutive points $\mathbf{p}^o \in \mathcal{L}_i$ in low-dimensional space[1]. The sequence of points included in $\mathcal{Q}_m$ is denoted by

$$\mathcal{Q}_m = \mathbf{p}^o_{r:v} = \{\mathbf{p}^o_r, \mathbf{p}^o_{r+1}, \ldots, \mathbf{p}^o_v\} \tag{1}$$

with $r, v \in \{1, \ldots, T\}$ and $r \leq v$. Here, $\mathbf{p}^o$ denotes poses of the observed agent, T is the total number of points in the trajectory, while M is the number of segments. The M segments are created by finding a set of sub-principal components that minimize the cost function

$$\Phi_m = \frac{1}{v - r + 1} \sum_{j=r}^{v} \mathbf{p}^o_j \mathbf{p}^{oT}_j \;\; , \; \mathbf{p}^o_j \in \mathcal{Q}_m. \tag{2}$$

In addition to allowing for varying joint correlations (and, thus, enabling the generation of IMs with varying topologies, see below), the segmentation of local posture spaces has the advantage of lowering the computational load of performing optimizations (see section V).

B. Global Posture Space, GMM, and HMM

A global posture space $\mathcal{G}$ is used to identify the relevant interaction demonstration and its associated local posture space $\mathcal{L}_i$. The global posture space is constructed from all recorded motions of the observed agent during all demonstrations by applying PCA to the motion capture data.

When databases with several interaction examples are considered, situations may arise in which the correct interaction example can only be determined when the temporal context is accounted for. To tackle this problem, we identify the context by using an HMM. To train the HMM we first employ a GMM density estimator as an initialization scheme. This initialization is performed via moment-matching [17], which also automatically determines the number K of kernels needed. As argued by [18], using kernel density estimation (in our case GMM) to extract key poses from continuous motion data often results in good approximations of the demonstrated motion.

[1]For notational clarity we denote the controlled agent $(\cdot)^c$ and the observed agent $(\cdot)^o$. Also, the transformations of a pose $\mathbf{p}$ into $\mathcal{G}$, $\mathcal{L}_i$ and $\mathbb{R}^3$ are achieved by a single matrix operation and we henceforth omit a precise marking in favour of readability. Instead we reference the corresponding coordinate system at each occurrence.

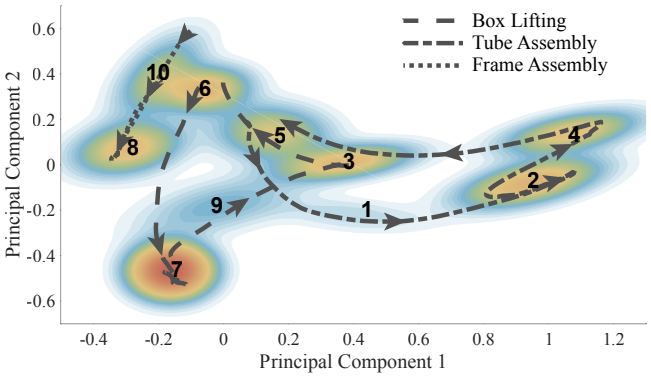

Fig. 3. A global posture space is created by applying PCA to the observed agents motion capture data. The resulting motion trajectories in latent space as well as the computed kernel density estimate ($K = 10$) is visualized for the tube assembly task. Color is used to indicate the probability of each distribution.

Using the extracted distributions (see Fig. 3) as a discrete set of key poses, we can now train a HMM that models the probability distribution over observed key pose sequences. Following the notation of [19], an HMM can be defined as a tuple $\Theta = (S, P_i, P_{i \to j}, p_i(o))$. S is a set of states s_i, $i, j \in \{1, \ldots, |S|\}$, of the hidden Markov model. P_i is a probability distribution of the probability starting in state s_i. $P_{i \to j}$ denotes the state transition matrix. Elements of $P_{i \to j}$ define the probability $p(s_j|s_i)$ of transitioning from state s_i to s_j. $p_i(o)$ is the emission probability distribution, which defines the probability of observing o while in state s_i.

We define a hidden state S for each interaction (subtask) and the index n of the most likely distribution of the mixture model for a pose $\mathbf{p}^o_t$ of the observed agent at time step $t \in \{1, \ldots, T\}$ an observation $o \in \{1, \ldots, K\}$

$$n = \arg\max_{\forall i} \mathcal{N}_i(\mathbf{p}^o_t | \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \; , i \in \{1, \ldots, K\}. \tag{3}$$

Here, $\mathcal{N}_i$ is the i-th Gaussian distribution with mean $\boldsymbol{\mu}_i$ and covariance $\boldsymbol{\Sigma}_i$.

Training data for the HMM is generated by computing the probability of each distribution at time step t. When the most probable distribution differs from the previous time step, a new key pose is appended to the key pose sequence. The key poses, denoted by their indices $\mathbf{o}$, are used as input (observations) to the HMM. Since we record each interaction separately, an estimate of the transition matrix $P_{i \to j}$ and the emission distribution $p(\mathbf{o})$ are computed by evaluating each occurrence of $p(s_j|s_i)$. The resulting HMM encodes sequences of key poses in the global posture space and it is

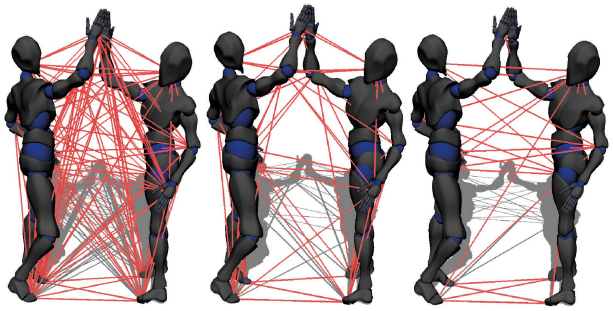

Fig. 4. Left: An IM computed using a fully connected graph. Middle: IMs created using Delaunay triangulation. Right: Using our method connections are added based on joint correlations in motion capture recordings. As it can be seen sparse topologies are created which allow for more joint movement during optimization and, consequently, increasing postural generalization.

therefore used to identify suitable interaction demonstrations, i.e. local posture spaces, during runtime.

C. Generating Correlation-based Interaction Meshes

Spatial adaptation and coordination using IMs has been demonstrated extensively by the computer animation community [2], [4], [15]. One eminent feature of IMs is the ability to adapt *full body* behaviors to new situations. So far, however, proposed methods rely on fully connected graphs [2] or Delaunay tetrahedralization [15] for net generation. These approaches include all joints equally into the topology yielding densely interconnected nets as shown in Fig. 4, left. Additional vertices are sampled on the skeleton's surface and increase the overall amount further, cf. [1]. Since the computational complexity of IM adaption significantly increases with larger numbers of vertices, only a few vertices should be used to ensure optimal response times of a robot. Moreover, joint weights and optimization constraints such as foot or hand contacts are usually modelled manually, thus requiring intervention of a human editor in the IM generation process. We propose the following extensions to Interaction Meshes:

- correlation-based topology generation at frame level for varying correlation structures and sparse marker setups
- data-driven optimization of weights to avoid manual labeling, and,
- an algorithm for automatic soft- and hard constraint generation based on motion capture data.

In our approach, an IM provides a topological and spatial representation of two humans during a motion capture recording at each time step. An IM topology is constructed using the Cartesian coordinates $\mathbf{x}^o$ and $\mathbf{x}^c$ of the closest pairs of motion capture markers u and l at each timestep t

$$(u,l)_t = \underset{\forall u,\forall l}{\operatorname{argmin}} \|\mathbf{x}_u^o - \mathbf{x}_l^c\| \quad u \in \{1,\ldots,N^o\},\ l \in \{1,\ldots,N^c\} \quad (4)$$

and their correlating neighboring markers j_1, j_2 of the controlled agent. Markers that exhibit correlation and are in close proximity are connected through a tedrahedron. A tetrahedron $\mathcal{T}_t$ is defined as a tuple of connected vertices $\mathcal{T}_{t,i} = (u,l,j_1,j_2)$ where u is an index into the motion capture marker table of the observed agent N^o and l,j_1 and j_2 correspond to the controlled agent's motion capture markers N^c. The topology of an IM $\mathcal{M}_t$ at time step t is consequently defined as a set of tetrahedrons $\mathcal{M}_t = \{\mathcal{T}_{t,1},\ldots,\mathcal{T}_{t,i},\ldots,\mathcal{T}_{t,R}\}$, where R denotes the amount of created tetrahedrons.

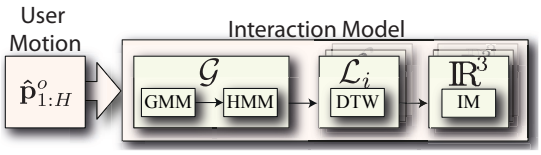

Fig. 5. During runtime the observed agent's motion $\hat{\mathbf{p}}_{1:H}$ is projected into the previously created global posture space $\mathcal{G}$. As each hidden state of the HMM is directly associated with a single interaction, the corresponding local posture space $\mathcal{L}_i$ can be retrieved and used to match the observed agent's motion onto motion segments. This yields a point from the initial demonstration that best fits the current situation temporally. To adapt the found reference pose spatially to the new situation the associated IM is optimized.

To allow for varying joint relationships during runtime a correlation-based weight matrix

$$W_m = 1 - \frac{|E[(\mathbf{p}_t^c - \boldsymbol{\mu}_m)(\mathbf{p}_{t+1}^c - \boldsymbol{\mu}_m)]|}{\sigma^2(\mathbf{p}_m^c)}\ ,\ \mathbf{p}^c \in \mathbb{R}^{N^c} \quad (5)$$

is computed for each segment $\mathcal{Q}_m$ (see Eq. 1). In Eq. 5 $\boldsymbol{\mu}_m$ denotes the mean and $\sigma^2(\mathbf{p}_m^c)$ the variance over poses $\mathbf{p}_m^c$ of segment $\mathcal{Q}_m$ in Cartesian coordinates.

Using the weight matrices, each marker of the controlled agent that does not exhibit a strong correlation is assigned a soft positional constraint according to its weights in W_m for each segment $\mathcal{Q}_m$. Additionally, we add a hard positional constraint for each joint of the observed agent, as the observed agent's pose is not subject to optimization.

V. Generating Online Robot Responses

To generate continuous robot responses during human-robot interaction, the interaction model is utilized in a sequential fashion (see Fig. 5). First, each live pose of the observed agent is reduced in dimensionality by projecting it in the global posture space $\mathcal{G}$, creating new points $\hat{\mathbf{p}}^o_{1:H} = \{\hat{\mathbf{p}}^o_1,\ldots,\hat{\mathbf{p}}^o_H\}$ with $\hat{\mathbf{p}}^o \in \mathcal{G}$, where H is an index to the most recent pose in the sliding window of evaluated poses.

Evaluating the posterior probability of each Gaussian distribution yields the most likely key poses for the user (see equation 3). The most suitable interaction is inferred by computing the posterior state probabilities of the HMM. Since each state s_i of the HMM is associated with a interaction demonstration, the corresponding local posture space $\mathcal{L}_i$ is retrieved and the human motion is matched against the segments $\mathcal{Q}_{1:M} = \{\mathcal{Q}_1,\ldots,\mathcal{Q}_M\}$ using Euclidean distances. This yields a segment $\hat{\mathcal{Q}}_m$ that best fits the current situation. A temporally suitable time step $\hat{t}$ of the initial recording is found by computing a DTW path [20] between the matched segment $\hat{\mathcal{Q}}_m$ and $\hat{\mathbf{p}}^o_{1:H}$. Given $\hat{t}$, an associated pair of poses $\mathbf{p}_{\hat{t}} = [\mathbf{p}^o_{\hat{t}}, \mathbf{p}^c_{\hat{t}}]$ from the initial recording, the corresponding IM $\mathcal{M}_{\hat{t}}$, the weight matrix $W_{\hat{t}}$ and a set of constraints is be retrieved.

Consider the difference between poses $\mathbf{p}_{\hat{t}}$ from the training recording with the poses in the current situation $\hat{\mathbf{p}}_H = [\hat{\mathbf{p}}^o_H, \hat{\mathbf{p}}^c_{current}]$, where $\hat{\mathbf{p}}^o_H$ and $\hat{\mathbf{p}}^c_{current}$ are the postures of the observed and controlled agent in Cartesian space.

In order to adapt the retrieved IM $\mathcal{M}_{\hat{t}}$ to the current situation, essentially, its deformation energy is minimized

$$\min_{\hat{\mathbf{p}}_H} \sum_{i=1}^{N^c+N^o} \frac{1}{2}\|L(\hat{\mathbf{p}}_{H,i}) - L(\mathbf{p}_{\hat{t},i})\|^2 + \sum_{i=1}^{N^c} W_{i,l}\|\hat{\mathbf{p}}_{H,i} - \mathbf{p}_{\hat{t},i}\|^2 \tag{6}$$

while at the same time ensuring the validity of its associated hard constraints. L is the Laplacian operator which deforms a pair of poses into local coordinates using the topology $\mathcal{M}_{\hat{t}}$.

The interested reader is referred to [1] and [4] for a more detailed description of the underlying optimization problem. In our implementation, however, IM topologies are computed at every frame t in order to create the robot behavior based on the current user pose allowing for context-sensitive and instant responses. At the same time, animation constraints are updated at the segment level which allows for varying joint weights during the course of an interaction.

VI. Evaluation

We evaluate our interaction learning method in two complex assembly tasks involving several manipulated objects. In the first example, a Lego rocket is collaboratively assembled with the help of a robot as shown in Fig. 6 and 9. In the second example, a tube frame is put together in collaboration between the user and a robot. During demonstration, a box containing a set of pipes is first placed on a stand and two pipes are assembled collaboratively, before the final tube frame is constructed. As illustrated in Fig. 9, the robot imitates the demonstrated motion successfully and the objects for both examples are assembled jointly with the user.

A. Experimental Setup

In general, we do not assume a specific motion capture hardware and have tested the system successfully with a marker-based optical tracking system[2] as well as a setup consisting of several Kinect depth sensors (see Appendix). In the considered human-robot collaboration tasks, however, marker-based tracking outperformed other capturing solutions in terms of reliability and accuracy and was, in consequence, used in the experiments described below.

In the following, we mainly focus on the Lego rocket assembly task. Similar results have also been obtained for the tube assembly, but are ommited for the sake of presentation and readability. In order to evaluate the generalization capabilities of our system, we recorded the rocket assembly task 14 times at different positions (see Fig. 6) resulting in 41000 motion capture frames (approx. 22 min). Using only a single demonstration to learn the interaction model, 13 repetitions of the joint task are available as validation data. Given these

[2]In our experiments we use tracking system by A.R.T. Each marker provides a position (and an unused orientation) at 30 Hz. During human-human demonstrations and human-robot interactions, each human wears six markers ($N^o = 6$, $N^c = 6$).

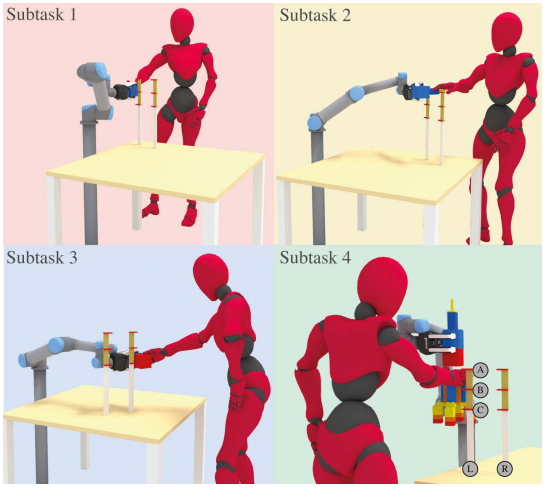

Fig. 6. The rendering shows 4 stages for the rocket assembly task. We recorded the joint interaction 14 times for 3 heights (A,B,C) at two positions (L,R) on the table each approximately 20 cm apart. For training our system we utilized a single demonstration with medium height B at location R.

as input, we compare a robot's response with that of a human interaction partner. Based on the training demonstration our preprocessing step created a significantly reduced state space $\mathcal{G}$ from initially 18 dimensions to 6 and 5 dimensions for the Lego rocket task and the tube assembly, respectively. The dimensionality of $\mathcal{L}_i$ of subtasks was 4 for both assembly tasks. Approximating the density of user motions in $\mathcal{G}$ was conducted with $K = 14$ and $K = 10$ kernels, yielding reliable results while compressing the motion to the most relevant key poses (see Fig. 3 for the tube assembly task). Segment matching as well as temporal pose matching using DTW in $\mathcal{L}_i$ was performed with a pose history of $H = 30$, resulting in a temporal history of approximately one second. This differs from other approaches where each pose is optimized based on a single observation and, in doing so temporal and contextual importances of joints are neglected.

B. Interaction Selection

Selecting an appropriate interaction template from the pool of all interactions is a crucial step of our system. During runtime the user's motion is reduced in dimensionality and the corresponding key postures are inferred based on Gaussian distributions. As a result a list of key poses of the initial recording are created that resemble how the user moved in the current task. The confusion matrix in Tab. I compares the classification accuracy using HMMs with an approach using Euclidean distances as employed in [1] and [16]. As indicated the overall number of false classifications is significantly higher using the approach in [1], [16].

Poses at the beginning and the end of each subtask are similar (both arms resting aside) and without incorporating past poses as context, reliable selections of the current active interaction are hard to generate. Using the HMM, sequential information of key poses is inherently incorporated and, thus, allows template selection on a broader contextual level. This, in turn, has a strong influence on the robot's hysteresis, i.e. the robot's tendency to remain committed to an interaction

TABLE I

CONFUSION MATRIX OF THE PROPORTION OF CORRECT GUESSES USING **HMMs** AND *Euclidean distances (DSTC)*.

Predicted Class \ Actual Class	Subtask 1 HMM / *DSTC*	Subtask 2 HMM / *DSTC*	Subtask 3 HMM / *DSTC*	Subtask 4 HMM / *DSTC*
Subtask 1	**0.94** *0.15*	**0.07** *0.42*	**0.04** *0.29*	**0.00** *0.38*
Subtask 2	**0.00** *0.00*	**0.93** *0.05*	**0.00** *0.15*	**0.00** *0.42*
Subtask 3	**0.00** *0.22*	**0.00** *0.18*	**0.96** *0.27*	**0.06** *0.25*
Subtask 4	**0.06** *0.64*	**0.00** *0.35*	**0.00** *0.29*	**0.94** *0.05*

task where the current pose of the human by itself might indicate a different interaction task (see Tab. I).

C. Spatial Generalization

Spatial generalization was evaluated for the Lego assembly task using one motion captured demonstration as well as additional 13 motion capture recordings of human-human interactions as validation data. Between the 13 task executions, the handover positions of the manipulated object were varied both in height (see Fig. 6 A,B,C) and location (L,R). In a simulation environment, the recorded motions of the observed agent were applied to a simulated human while the responses of a simulated robot were computed using our interaction model. The simulated robot's motions were then compared to the motions of the human assistant in the validation cases. Fig. 8 depicts the robot's response in three executions of the assembly tasks including all substasks. In the figure, blue trajectories depict the height of the human hand during validation while red trajectories show hand hight in the demonstration used to train the interaction model. The robot's adapted gripper height is shown in green. In almost all examples, the robot optimized its position to match the human's hand and reached heights similar to the validation data. However, in some situations spatial adaptation was insufficient. E.g. in execution 1, subtask 4, the robot only adapted to a height of about 19 cm where 27 cm would have been required. In all other examples, the robot adapted its behavior so that the interaction could successfully be completed, i.e. the object could be jointly assembled.

Fig. 7 illustrates the variance in spatial generalization for different net topology generation methods. Compared to alternative topologies, our method exhibits the largest variance and, thus, offers spatial generalization to a larger range of positions. The deformation of a found reference IM $\mathcal{M}_{\hat{t}}$ before optimization was on average 0.25. After adapting the IM to the current situation the deformation energy is reduced to about 0.023. In contrast to IMs created with Delaunay tetrahedralization, the deformation is reduced to 0.1. However, since traditional IMs do not focus on important joints, varying user torso rotations force the reconstruction to adapt and change robot hand positions (-10 cm to 10 cm) even when user hand positions do not change.

Further, situations emerged where adaptation of more than 10 cm was required, i.e. height A and C in Fig. 6. Here, alternative interaction mesh topologies did not provide the required degree of adaptation prompting the user to adapt

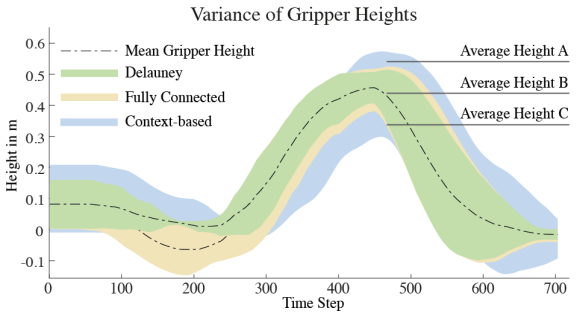

Fig. 7. The figure shows heights that can be generalized to using IMs with different topology generation methods. Delauney triangulation (green area) only allows hand-overs differing up to 10 cm in height. Using our data-driven topology generation scheme (blue area) context-sensitive edges and weights are computed. These allow for much wider postural generalization of up to -25 cm to 25 cm. Interestingly, fully connected topologies (yellow area) provide better generalization than nets created with Delauney triangulation.

to the robot instead. This unnecessarily requires users to match the original motion capture recording closely, in order to interact with the robot successfully. Our method on the other hand weights each marker based on their contribution to the overall motion and adapts only relevant joints. This allows users a broader range of movement and a more natural interaction.

D. Temporal Generalization

Temporal generalization is achieved using a two stage process. First, the user's motion is matched against interaction templates using the HMM and, then, aligned locally in $\mathcal{L}_i$ using DTW. Fig. 8 shows three repetitions of the Lego assembly task with varying execution speeds, with a time difference between the slowest and fastest task completion of $\sim$ 20 s. The actual execution time in the validation cases is indicated by the blue trajectory. As can be seen in the Figure, the shapes of the red trajectory (selection of an appropriate IM) and the green trajectory (adaption of the selected IM to the current situation) closely match the shape of the blue trajectory. This shows that our method is able to maintain a close temporal synchrony between the movements of the human and the robot even if the task execution time is quite different from the training example.

E. Performance

As our variant of IMs uses a sparse set of motion capture markers instead of a comprehensive set of human joints as vertices, an inverse kinematics solver must be employed to compute the joint angles of the robot's target pose. Using inverse kinematics solvers circumvents the challenge of mapping human data onto a robot (the correspondence problem) and ensures that joint limits are not violated.

On average, computing the final configuration of the robot using inverse kinematics accounts for 50 % of the computation time (16 ms) where as inferring an interaction and computing suitable responses using our context-based IMs requires an additional 17 ms. Using our methodology the robot is continuously controlled with a latency of approximately 150 ms towards observed user poses.

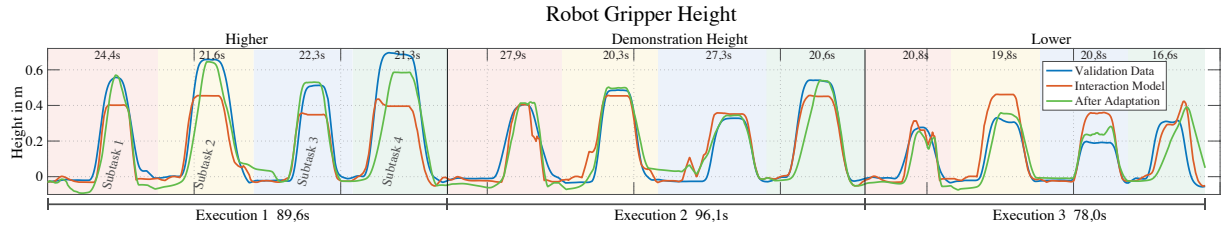

Fig. 8. The figure shows 3 variations of the Lego assembly task with differing hand-over heights. On the left, input user motions significantly higher than the original recording are shown. In the middle hand-over heights similar to the training data and on the right lower hand-over heights are illustrated. The figure depicts how a frame from the interaction model (red trajectory) is adapted to the new situation (blue trajectory). The green trajectory shows gripper heights after our adaptation. Our system successfully generalizes up to (-25 cm to 25 cm) in height, outperforming approaches with conventional, not correlation-based interaction mesh topologies by a large margin.

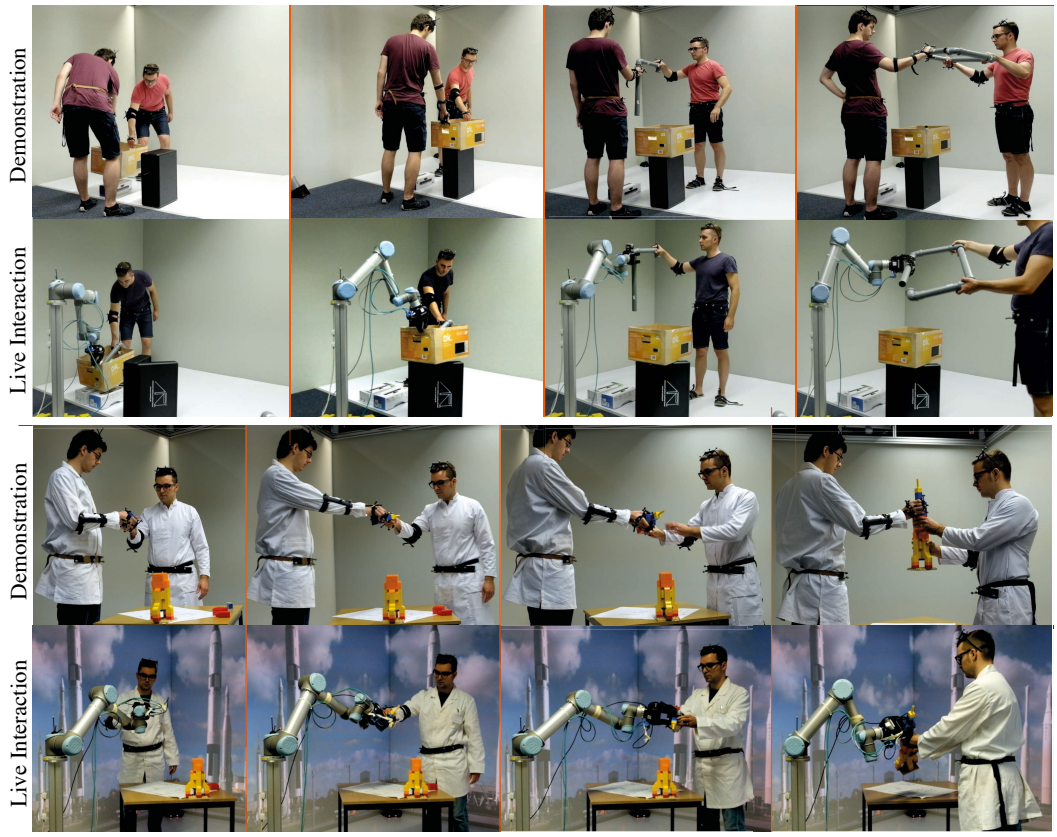

Fig. 9. Top half: A tube frame is assembled with a user. Top row: Recording of human-human interactions using optical motion capture. In our experiments we record three interactions, lifting a box, extracting and assembling a tube before finally constructing a pipe frame. Bottom row. Our method is utilized during human-robot interaction and the robotic arm is continuously reacting to observed postures. Lower half: In a second scenario, a Lego rocket consisting of four parts is assembled with a user. The figure shows that the robot imitates the demonstrated behavior successfully while at the same time adopting its poses to the new situation.

VII. Discussion

Training interactive robots by providing human-human demonstration as demonstrated above is a promising approach towards the specification of interaction dynamics. Our experiments, however, have also revealed various technical challenges. At the moment, the availability of such human-human recordings is very limited, since most databases include only single person movements. Also, recording motion capture data with two humans is challenging, due to self-occlusions and line-of-sight problems.

The experimental results show a reasonable ability of learned interaction models to generalize to spatial and temporal changes. However, the system cannot deal with large spatial adaptations. In these cases, generalization to new positions cannot be achieved successfully without violating constraints. At the moment, spatial adaptations within -25 cm to 25 cm of the initial demonstration are feasible.

A general insight of our experiments is that human-robot interaction can greatly benefit from graph-based spatial representations. The relationship between two interaction partners can be modelled as a mesh between joints, which is then analyzed via the graph Laplacian, and other well-established graph-theoretic measures. We have also shown that the topology of the graph has a strong influence on the generalization capabilities of demonstrated tasks. Defining

correlations among joints explicitly in an IM has proven to yield good generalization results, if the topology is chosen carefully.

The opening and closing of the hands is currently not captured during human-human demonstrations and we therefore control the robot gripper manually. However, this could potentially be modeled as a binary emission within the HMM, a research direction we are currently investigating.

VIII. Conclusion

We presented a learning by demonstration methodology that enables a robotic arm to seamlessly interact with a human in two collaborative tasks. A novel aspect is that our method is based on motion captured recordings of parallel behavior demonstrations by two human interaction partners. In contrast e.g. to approaches based on kinesthetic training, the learned interaction models inherently capture the important spatial relationships and temporal synchrony of body movements between two interacting partners. This enables the robot to continuously adapt its behavior, both spatially and temporally, to the ongoing actions of the human interaction partner. Therefore, the presented approach is well suited for collaborative tasks requiring continuous body movement coordination of a human and a robot.

IMs were originally proposed in the context of non real-time computer animation. In order to apply them to real-time human-robot interaction settings, we developed a variant with comparably few vertices. This not only decreases computational demands for mesh optimization but also improves the spatial generalization capabilities.

A current drawback of our method is, that an inverse kinematics solver is needed to generate the final robot pose from the more abstract human pose extracted from the IM. However, since our method is fundamentally based on the reconstruction of a humanoid skeletal structure it has the potential of being applicable to anthropological robots as well. By changing the kinematic chain of the inverse kinematics solver one could create seamless controls for complex articulated humanoids as experiments in virtual reality suggest [4].

References

[1] E. S. L. Ho, T. Komura, and C. Tai, "Spatial relationship preserving character motion adaptation," *ACM Trans. Graph.*, vol. 29, no. 4, pp. 33:1–33:8, 2010.

[2] V. Ivan, D. Zarubin, M. Toussaint, T. Komura, and S. Vijayakumar, "Topology-based representations for motion planning and generalization in dynamic environments with interactions," *The International Journal of Robotics Research*, vol. 32, no. 9-10, pp. 1151–1163, 2013.

[3] Y. Yang, V. Ivan, and S. Vijayakumar, "Real-time motion adaptation using relative distance space representation," in *2015 International Conference on Advanced Robotics (ICAR)*, no. 17. IEEE, July 2015, pp. 21–27.

[4] D. Vogt, B. Lorenz, S. Grehl, and B. Jung, "Behavior generation for interactive virtual humans using context-dependent interaction meshes and automated constraint extraction," *Computer Animation and Virtual Worlds*, vol. 26, no. 3-4, pp. 227–235, May 2015.

[5] T. Inamura, I. Toshima, H. Tanie, and Y. Nakamura, "Embodied Symbol Emergence Based on Mimesis Theory," *The International Journal of Robotics Research*, vol. 23, no. 4, pp. 363–377, apr 2004.

[6] D. Kulic and Y. Nakamura, "Scaffolding on-line segmentation of full body human motion patterns," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, Sep 2008, pp. 2860–2866.

[7] T. Ravet, J. Tilmanne, and N. D'Alessandro, "Hidden Markov Model Based Real-Time Motion Recognition and Following," in *Proceedings of the 2014 International Workshop on Movement and Computing - MOCO '14*. New York, New York, USA: ACM Press, 2014, pp. 82–87.

[8] H. Ben Amor, D. Vogt, M. Ewerton, E. Berger, B. Jung, and J. Peters, "Learning responsive robot behavior by imitation," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, Nov. 2013, pp. 3257–3264.

[9] P. Pastor, H. Hoffmann, T. Asfour, and S. Schaal, "Learning and generalization of motor skills by learning from demonstration," in *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, May 2009, pp. 763–768.

[10] B. Argall, S. Chernova, M. M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robotics and Autonomous Systems*, vol. 57, no. 5, pp. 469–483, 2009.

[11] H. Ben Amor, "Imitation Learning of Motor Skills for Synthetic Humanoids," Ph.D. dissertation, Technische Universität Bergakademie Freiberg, 2010.

[12] L. Rozo, D. Bruno, S. Calinon, and D. G. Caldwell, "Learning optimal controllers in human-robot cooperative transportation tasks with position and force constraints," in *Proc. IEEE/RSJ Intl Conf. on Intelligent Robots and Systems (IROS)*, 2015, pp. 1024–1030.

[13] M. Ewerton, G. Neumann, R. Lioutikov, H. Ben Amor, J. Peters, and G. Maeda, "Learning Multiple Collaborative Tasks with a Mixture of Interaction Primitives," in *Proceedings of the International Conference on Robotics and Automation (ICRA)*, 2015.

[14] D. Lee, C. Ott, and Y. Nakamura, "Mimetic Communication Model with Compliant Physical Contact in Human–Humanoid Interaction," *The International Journal of Robotics Research*, vol. 29, no. 13, pp. 1684–1704, Nov 2010.

[15] E. S. L. Ho, J. C. P. Chan, T. Komura, and H. Leung, "Interactive Partner Control in Close Interactions for Real-time Applications," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 9, no. 3, pp. 21:1—-21:19, July 2013.

[16] D. Vogt, S. Grehl, E. Berger, H. B. Amor, and B. Jung, "A data-driven method for real-time character animation in human-agent interaction," in *Intelligent Virtual Agents - 14th International Conference, IVA 2014, Boston, MA, USA, August 27-29, 2014. Proceedings*, ser. Lecture Notes in Computer Science, T. W. Bickmore, S. Marsella, and C. L. Sidner, Eds., vol. 8637. Springer, 2014, pp. 463–476.

[17] M. Kristan, A. Leonardis, and D. Skočaj, "Multivariate online kernel density estimation with Gaussian kernels," *Pattern Recognition*, vol. 44, no. 10-11, pp. 2630–2642, Oct. 2011.

[18] B. Akgun, M. Cakmak, K. Jiang, and A. L. Thomaz, "Keyframe-based Learning from Demonstration," *International Journal of Social Robotics*, vol. 4, no. 4, pp. 343–355, Nov 2012.

[19] M. Brand and A. Hertzmann, "Style Machines," *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pp. 183–192, 2000.

[20] D. J. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series," in *Knowledge Discovery in Databases, Seattle, Washington*, 1994, pp. 359–370.

Appendix

A high-resolution video demonstrating the proposed method on the tube assembly task can be found at `https://youtu.be/_2qcU4FcGyE`.

Also, the interested reader is encouraged to view our "popular science"-oriented video showcasing the Lego assembly task `https://youtu.be/P7utkiVhU-I`.

A video showcasing our methodology in a handover task using multiple *Kinect V2* sensors can be found at `https://youtu.be/KhcvUUO-ZE0`.