Learning Social Affordance Grammar from Videos: Transferring Human Interactions to Human-Robot Interactions

Tianmin Shu[1], Xiaofeng Gao[2], Michael S. Ryoo[3] and Song-Chun Zhu[1]

Abstract— In this paper, we present a general framework for learning social affordance grammar as a spatiotemporal AND-OR graph (ST-AOG) from RGB-D videos of human interactions, and transfer the grammar to humanoids to enable a real-time motion inference for human-robot interaction (HRI). Based on Gibbs sampling, our weakly supervised grammar learning can automatically construct a hierarchical representation of an interaction with long-term joint sub-tasks of both agents and short term atomic actions of individual agents. Based on a new RGB-D video dataset with rich instances of human interactions, our experiments of Baxter simulation, human evaluation, and real Baxter test demonstrate that the model learned from limited training data successfully generates human-like behaviors in unseen scenarios and outperforms both baselines.

I. INTRODUCTION

With the recent progress in robotics, robots now have been able to perform many complex tasks for humans. As a result, it is inevitable that the robots will interact with humans in various social situations, such as service robots taking care of elderly people, robot co-workers collaborating with humans in a workplace, or simply a robot navigating through human crowds. Similar to human social interactions, human-robot interactions (HRI) must also follow certain social etiquette or social norms, in order to make humans comfortable.

Conventional robot task planing only consider the effectiveness and efficiency of performing specific tasks, such as manufacturing, cleaning, and other activities that do not consider human values or preference. However, as J. J. Gibson pointed out, "The richest and most elaborate affordances of the environment are provided by ... other people." [1]. A robot should reason the intention and feeling of humans who are near it and only perform socially appropriate actions while trying to achieve its own goal.

Therefore, in this paper, we focus on learning social affordances in human daily activities, namely action possibilities following basic social norms, from human interaction videos. More specifically, we are interested in the following three general types of human-robot interactions that we believe are most dominant interactions for robots: i) social etiquette, e.g., greeting, ii) collaboration, e.g., handing over objects, and iii) helping, e.g., pulling up a person who falls down. In addition, we also aim at developing a real-time motion inference to

[1]Center for Vision, Cogntion, Learning, and Autonomy, University of California, Los Angeles, USA `tianmin.shu@ucla.edu`, `sczhu@stat.ucla.edu`
[2]Department of Electronic Engineering, Fudan University, China `xfgao13@fudan.edu.cn`
[3]School of Informatics and Computing, Indiana University, Bloomington, USA `mryoo@indiana.edu`
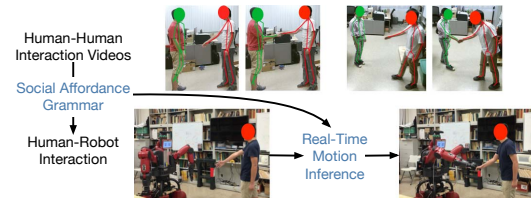
Fig. 1. The framework of our approach.

enable natural human-robot interactions by transferring the social affordance grammar.

To this end, we propose a new representation for social affordances, i.e., social affordance grammar as a spatiotemporal AND-OR graph (ST-AOG), which encodes both important latent sub-goals for a complex interaction and the fine grained motion grounding such as human body gestures and facing directions. We learn the grammar from RGB-D videos of human interactions as Fig. 1 depicts. Our grammar model also enables short-term motion generation (e.g., raising an arm) for each agent independently while providing long-term spatiotemporal relations between two agents as sub-goals to achieve for both of them (e.g., holding the right hand of each other), which simultaneously maximizes the flexibly of our motion inference (single agent action) and grasps the most important aspects of the intended human-robot interactions (sub-goals in joint tasks).

Contributions:
1) A general framework for weakly supervised learning of social affordance grammar as a ST-AOG from videos;
2) A real-time motion inference based on the ST-AOG for transferring human interactions to HRI.

II. RELATED WORK

Affordances. In the existing affordance research, the domain is usually limited to object affordances [2], [3], [4], [5], [6], [7], [8], [9], e.g., possible manipulations of objects, and indoor scene affordances [10], [11], e.g., walkable or standable surface, where social interactions are not considered. [12] is the first to propose a social affordance representation for HRI. However, it could only synthesize human skeletons rather than control a real robot, and did not have the ability to generalize the interactions to unseen scenarios. We are also interested in learning social affordance knowledge, but emphasize on transferring such knowledge to a humanoid in a more flexible setting.

Structural representation of human activities. In recent years, several structural representations of human activities for the recognition purposes have been proposed for human action recognition [13], [14], [15], [16] and for group activity
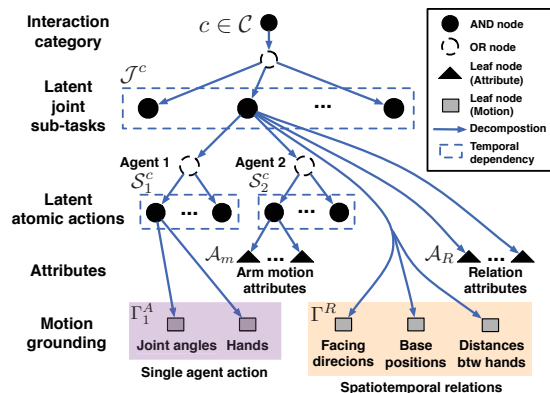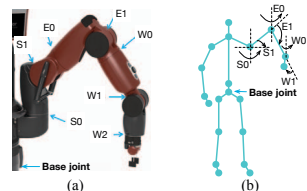
Fig. 2. Social affordance grammar as a ST-AOG.

Fig. 3. (a) The joint angles of the arm of a Baxter robot (from http://sdk.rethinkrobotics.com/wiki/Arms), which are directly mapped to a human's arm (b). The additional angles (e.g., w_2) can be either computed by inverse kinematics or set to a constant value.

recognition [17], [18], [19], [20], [21], [22], [23]. There also have been studies of robot learning of grammar models [24], [25], [26], but they were not aimed for HRI.

Social norms learning for robots. Although there are previous works on learning social norms from human demonstrations aimed for robot planning, they mostly focused on relatively simple social scenarios, such as navigation [27], [28]. On the contrary, we are learning social affordances as a type of social norm knowledge for much more complex interactions, which involve the whole body movements.

III. FRAMEWORK OVERVIEW

The framework of our approach illustrated in Fig. 1 can be outlined as follows:

Human videos. We collect RGB-D videos of human interactions, where human skeletons were extracted by Kinect. We use the noisy skeletons of these interactions as the input for the affordance learning.

Social affordance grammar learning. Based on the skeletons from human interaction videos, we design a Gibbs sampling based weakly supervised learning method to construct a ST-AOG grammar as the representation of social affordances for each interaction category.

Real-Time motion inference. For transferring human interactions to human-robot interactions, we propose a real-time motion inference algorithm by sampling parse graphs as hierarchical plans from the learned ST-AOG and generate human-like motion accordingly for a humanoid to interact with a human agent.

IV. REPRESENTATION

We represent the social affordance knowledge as stochastic context sensitive grammar using a spatiotemporal AND-OR graph (ST-AOG), as shown in Fig. 2. The key idea is to

model the joint planning of two agents on top of independent action modeling of individual agents. Following the Theory of Mind (ToM) framework, a ST-AOG defines the grammar of possible robotic actions (agent 2) at a specific moment given the observation of agent 1's actions as the belief, the joint sub-tasks as sub-goals, and the interaction category as the overall goal.

We first define a few dictionaries for the grammar model encoding the key elements in the social affordances. We constrain the human-robot interactions in a set of categories $\mathcal{C}$. Dictionaries of arm motion attributes $\mathcal{A}_M$ and relation attributes $\mathcal{A}_R$ are specified and shared across all types of interactions. Also, for each category c, there are dictionaries of latent joint sub-tasks $\mathcal{J}^c$, latent atomic actions of agent i, $\mathcal{S}_i^c$, where $\mathcal{S}_i^c$ are shared by different joint sub-tasks within c. Note that joint sub-tasks and atomic actions are not predefined labels but rather latent symbolic concepts mined from human activity videos, which boosts the flexibility of our model and requires much less human annotation efforts.

There are several types of nodes in our ST-AOG: An AND node defines a production rule that forms a composition of nodes; an OR node indicates stochastic switching among lower-level nodes; the motion leaf nodes show the observation of agents' motion and their spatiotemporal relations; attribute leaf nodes provide semantics for the agent motion and spatiotemporal relations, which can greatly improve the robot's behavior. In our model, we consider four arm motion attributes, i.e., *moving left/right arm*, *static left/right arm*, and the relation attributes include *approaching* and *holding* between two agents' hands (possibly an object).

The edges $\mathcal{E}$ in the graph represent decomposition relations between nodes. At the top level, a given interaction category leads to a selection of joint sub-tasks as the sub-goal to achieve for the given moment. A joint sub-task further leads to the atomic action selection of two agents and can also be bundled with relation attributes. An atomic action encodes a consistent arm motion pattern, which may imply some arm motion attributes of agent 2 for the purpose of motion inference. Some of the nodes in the dashed box are connected representing the "followed by" relations between joint sub-tasks or atomic actions with certain transition probabilities.

The motion grounding is designed for motion transfer from a human to a humanoid, which entails social etiquette such as proper standing distances and body gestures. As shown in Fig. 3, the pose of a human arm at time t can be conveniently mapped to a robot arm by four degrees: $\theta^t = \langle s_0, s_1, e_0, e_1 \rangle$. The wrist angles are not considered due to the unreliable hand gesture estimation from Kinect. Thus, in an interaction whose length is T, there is a sequence of joint angles, i.e., $\Theta_{il} = \{\theta_{il}^t\}_{t=1,\cdots,T}$ for agent i's limb l, where $l = 1$ stands for left arm and $l = 2$ indicates right arm. Similarly the hand trajectories $H_{il} = \{\boldsymbol{h}_{il}^t\}$ are also considered in order to have a precise control of the robot's hands. We model the spatiotemporal relations with agent 2's the relative facing directions, $O = \{o^t\}_{t=1,\cdots,T}$, and relative base positions (in the top-down view), $X = \{\boldsymbol{x}^t\}_{t=1,\cdots,T}$, by setting the facing directions and base joint positions of
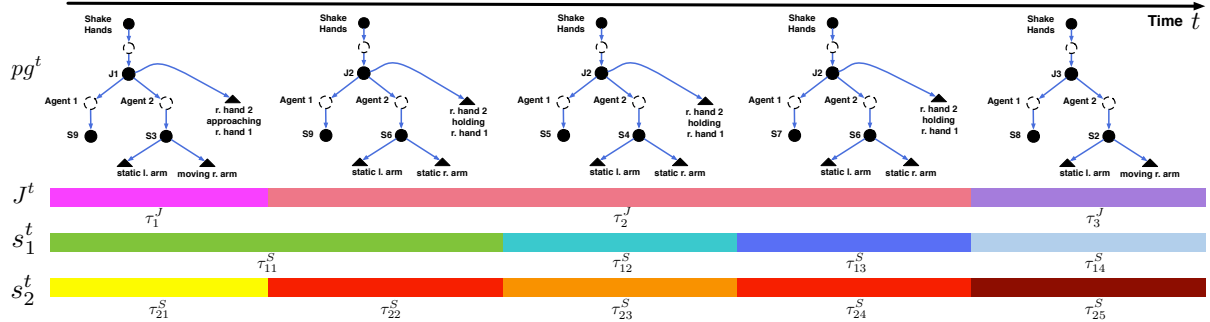
Fig. 4. A sequence of parse graphs in a shaking hands interaction, which yields the temporal parsing of joint sub-tasks and atomic actions depicted by the colored bars (colors indicate the labels of joint sub-tasks or atomic actions).

agent 1 as references respectively. We also consider the distances between two agents' hands, $D_{ll'} = \{d_{ll'}^t\}_{t=1,\cdots,T}$ (l is the limb of agent 1 and l' is the limb of agent 2) for the relations. The distances between agent 2's hands and an object can be included if an object is involved. For an interaction instance, we then define the action grounding of agent i to be $\Gamma_i^A = \langle \Theta \rangle$, and the relation grounding of both agents to be $\Gamma^R = \langle O, X, D \rangle$, where $\Theta = \{\Theta_{il}\}_{l=1,2}$, $H = \{H_{il}\}_{l=1,2}$, and $D = \{D_{ll'}\}_{l,l' \in \{1,2\}}$. Hence, the overall motion grounding is $\Gamma = \langle \{\Gamma_i^A\}_{i=1,2}, \Gamma^R \rangle$.

Finally, the ST-AOG of interactions $\mathcal{C}$ is denoted by $\mathcal{G} = \langle \mathcal{C}, \{\mathcal{J}^c\}_{c \in \mathcal{C}}, \{\mathcal{S}_i^c\}_{c \in \mathcal{C}, i=1,2}, \mathcal{A}_M, \mathcal{A}_R, \Gamma, \mathcal{E} \rangle$. At any time t, we use a sub-graph of the ST-AOG, i.e., a parse graph $pg^t = \langle c, j^t, s_1^t, s_2^t \rangle$, to represent the actions of individual agents (s_1^t, s_2^t) as well as their joint sub-tasks (j^t) in an interaction c. Note that the attributes are implicitly included in the parse graphs since they are bundled with labels of j^t and s_2^t.

For an interaction in $[1, T]$, we may construct a sequence of parse graphs $PG = \{pg^t\}_{t=1,\cdots,T}$ to explain it, which gives us three label sequences: $J = \{j^t\}_{t=1,\cdots,T}$, $S_1 = \{s_1^t\}$ and $S_2 = \{s_2^t\}$. By merging the consecutive moments with the same label of joint sub-tasks or atomic actions, we obtain three types of temporal parsing, i.e., $\mathcal{T}^J = \{\tau_k^J\}_{k=1,\cdots,K^J}$, $\mathcal{T}_1^S = \{\tau_{1k}^S\}_{k=1,\cdots,K_1^S}$, and $\mathcal{T}_2^S = \{\tau_{2k}^S\}_{k=1,\cdots,K_2^S}$ for the joint sub-tasks and the atomic actions of two agents respectively, each of which specifies a series of consecutive time intervals where the joint sub-task or the atomic action remains the same in each interval. Hence, in $\tau_k^J = [t_k^1, t_k^2]$, $j^t = j(\tau_k^J)$, $\forall t \in \tau_k^J$, and for agent i, $s_i^t = s_i(\tau_{ik}^S)$, $\forall t \in \tau_{ik}^S$ in $\tau_{1k}^S = [t_{ik}^1, t_{ik}^2]$. Fig. 4 shows an example of the temporal parsing from the parse graph sequence. Note the numbers of time intervals of these three types of temporal parsing, i.e., K^J, K_1^S, and K_2^S, may be different. Such flexible temporal parsing allows us to model long-term temporal dependencies among atomic actions and joint sub-tasks.

V. PROBABILISTIC MODEL

We propose a probabilistic model for our social affordance grammar model.

Given the motion grounding, Γ, the posterior probability of a parse graph sequence PG is defined as

$$p(PG|\Gamma) \propto \underbrace{p(\{\Gamma_i^A\}_{i=1,2}|PG)}_{\text{arm motion likelihood}} \underbrace{p(\Gamma^R|PG)}_{\text{relation likelihood}} \underbrace{p(PG)}_{\text{parsing prior}}. \quad (1)$$

Conditioned on the temporal parsing of atomic actions and joint sub-tasks, the likelihood terms model the arm motion and the relations respectively, whereas the parsing prior models the temporal dependencies and the concurrency among joint sub-tasks and atomic actions. We introduce these three terms in the following subsections.

A. Arm Motion Likelihood

First, we define three types of basic potentials that are repeatedly used in the likelihood terms:

1) **Orientation potential** $\psi_o(\theta)$. This potential is a von Mises distribution of the orientation variable θ. If θ has multiple angular variables, e.g., the four joint angles $\theta = \langle s_0, s_1, e_0, e_1 \rangle$, then the potential is the product of the von Mises distributions of these individual angular variables.

2) **Three-dimensional motion potential** $\psi_{3v}(\boldsymbol{x})$. Assuming that spherical coordinate of $\boldsymbol{x}$ is (r, θ, ϕ), the potential is characterized by three distributions, i.e., $\psi_{3v}(\boldsymbol{x}) = p(r)p(\theta)p(\phi)$, where the first one is a Weibull distribution and the remaining are von Mises distributions.

3) **Two-dimensional position potential** $\psi_{2v}(\boldsymbol{x})$. We fit a bivariate Gaussian distribution for $\boldsymbol{x}$ in this potential.

For joint angles and hand positions in an atomic action, we are interested in their final statuses and change during the atomic action. Thus, for the limb l of agent i in the interval τ_{ik}^S assigned with atomic action $s_i(\tau_{ik}^S) \in \mathcal{S}^\rfloor$ such that $s_i^t = s_i(\tau_{ik}^S)$, $\forall t \in \tau_{ik}^S$, the arm motion likelihood

$$p(\Theta_{il}, H_{il}|\tau_{ik}^S, s_i(\tau_{ik}^S))$$
$$\propto \underbrace{\psi_o(\theta_{il}^{t'} - \theta_{il}^t)}_{\text{joint angles's change}} \underbrace{\psi_o(\theta_{il}^{t'})}_{\text{final joint angles}} \underbrace{\psi_{3v}(\boldsymbol{h}_{il}^{t'} - \boldsymbol{h}_{il}^t)}_{\text{hand movement}} \underbrace{\psi_{3v}(\boldsymbol{h}_{il}^{t'})}_{\text{final hand position}},$$
$$(2)$$

where $t = t_{ik}^1$ and $t' = t_{ik}^2$ are the starting and ending moments of τ_{ik}^S. Assuming independence between the arms, the arm motion likelihood for agent i in τ_{ik}^S is

$$p(\Gamma_i^A|\tau_{ik}^S, s(\tau_{ik}^S)) = \prod_l p(\Theta_{il}, H_{il}|\tau_{ik}^S, s_i(\tau_{ik}^S)), \quad (3)$$

and the arm motion likelihood for the entire interaction is

$$p(\Gamma_i^A|PG) = \prod_k p(\Gamma_i^A|\tau_{ik}^S, s(\tau_{ik}^S)). \quad (4)$$

Finally, the overall arm motion likelihood is the product of two agents' arm motion likelihood, i.e.,

$$p(\{\Gamma_i^A\}_{i=1,2}|PG) = \prod_i p(\Gamma_i^A|PG). \quad (5)$$

B. Relation Likelihood

Relation likelihood models the spatiotemporal patterns hidden in facing directions O, base positions X, and the distances between two agents' hands during a joint sub-task. In a interval τ_k^J with the same joint sub-task label $j(\tau_k^J)$ such that $j^t = j(\tau_k^J)$, $\forall t \in \tau_k^J$, the relation likelihood is

$$p(\Gamma^R | \tau_k^J, j(\tau_k^J)) \quad \propto \quad \underbrace{\psi_o(o^{t'})}_{\text{facing direction}} \underbrace{\psi_{2v}(\boldsymbol{x}^{t'})}_{\text{base position}}$$
$$\cdot \prod_{l,l'} \underbrace{\psi_{3v}(d_{ll'}^{t'})}_{\text{final hand distance}} \underbrace{\psi_{3v}(d_{ll'}^{t'} - d_{ll'}^{t})}_{\text{distance change}},$$

(6)

where τ_k^J starts at $t = t_k^1$ and ends at $t' = t_k^2$.

Hence, the overall relation likelihood can be written as

$$p(\Gamma^R | PG) = \prod_k p(\Gamma^R | \tau_k^J, j(\tau_k^J)). \tag{7}$$

C. Parsing Prior

The prior of a sequence of parse graphs is defined by the following terms:

$$p(PG) = \underbrace{\prod_k p\left(|\tau_k^J| \,\middle|\, j(\tau_k^J)\right)}_{\text{duration prior of joint sub-tasks}}$$
$$\cdot \underbrace{\prod_k p\left(|\tau_{1k}^S| \,\middle|\, s_1(\tau_{1k}^S)\right) \prod_k p\left(|\tau_{2k}^S| \,\middle|\, s_2(\tau_{2k}^S)\right)}_{\text{duration prior of atomic actions}}$$
$$\underbrace{\prod_{k>1} p\left(s_1(\tau_{1k}^S)|s(\tau_{1k-1}^S)\right)}_{\text{action transition for agent 1}} \underbrace{\prod_{k>1} p\left(s_2(\tau_{2k}^S)|s(\tau_{2k-1}^S)\right)}_{\text{action transition for agent 2}}$$
$$\cdot \underbrace{\prod_t p(s_1^t|j^t)p(s_2^j|j^t)}_{\text{concurrency}} \underbrace{\prod_{k>1} p\left(j(\tau_k^J)|j(\tau_{k-1}^J)\right)}_{\text{joint sub-task transition}},$$

(8)

where the duration priors follow log-normal distributions and the remaining priors follow multinomial distributions.

VI. LEARNING

The proposed ST-AOG can be learned in a weakly supervised manner, where we only specify the generic dictionaries of attributes and the sizes of the dictionaries of joint sub-tasks and atomic actions for each interaction. Given N training instances, $\Gamma = \{\Gamma_n\}_{n=1,\cdots,N}$, of an interaction category, where $\Gamma_n = \langle \{\Gamma_i^A\}_{i=1,2}, \Gamma_i^R \rangle$ is the motion grounding of instance n, the goal of learning is to find the optimal parsing graph sequence, PG_i, for each instance by maximizing the posterior probability defined in (1); then the ST-AOG is easily constructed based on the parse graphs.

It is intractable to search for the optimal parsing of atomic actions and joint sub-tasks simultaneously, which will take an exponential amount of time. Instead, we first 1) parse atomic actions for each agent independently and then 2) parse joint sub-tasks. Based on the likelihood distributions from the parsing results, we may 3) further obtain the implied attributes for each type of joint sub-tasks and atomic actions. We introduce the details in the rest of this section.
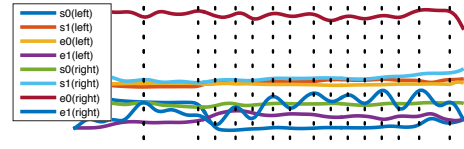

Fig. 5. The curves show how the joint angles of agent 2's two arms change in an shaking hands interaction. The black dashed indicate the interval proposals from the detected turning points.

A. Atomic Action Parsing

We expect the motion in an atomic action to be consistent. Since the arm motion is characterized by joint angles and hand positions, the velocities of joints and hand movements should remain the same in an atomic action. Following this intuition, we propose the time intervals for the atomic actions of an agent by detecting the turning points of the sequences of joint angles (see Fig. 5), which will naturally yields time intervals of atomic actions. To make the angles directly comparable, they are all normalized to the range of $[0, 1]$.

To detect such turning points, we introduce a entropy function for a sequence $\{x^t\}$, i.e., $\mathcal{E}(t, w)$, where t is the location of interest and w is the window size. To compute $\mathcal{E}(t, w)$, we first count the histogram of the changes between consecutive elements, i.e., $x^t - x^{t-1}$ in the sub-sequence $\{x^{t'}\}_{t'=t-w,,t+w}$, and then $\mathcal{E}(t, w)$ is set to be the entropy of the histogram. By sliding windows with different sizes ($w = 2, 5, 10, 15$), we may detect multiple locations with entropy that is higher than a given threshold. By non-maximum suppression, the turning points are robustly detected.

After obtaining the time intervals, we assign optimal atomic action labels to each interval by Gibbs sampling. At each iteration, we choose an interval τ and sample a new label s for it based on the following probability:

$$s \sim p(\Gamma_i^A \mid \tau, s)p(\tau, s \mid \mathcal{T}_i^S \backslash \tau, S_i \backslash \{s_i^t\}_{t \in \tau}). \tag{9}$$

Here, $p(\Gamma_i^A \mid \tau, s)$ is the likelihood in (3), and based on the parsing prior in (8), the labeling prior is computed as

$$p(\tau, s \mid \mathcal{T}_i^S \backslash \tau, S_i \backslash \{s_i^t\}_{t \in \tau}) = p(s \mid s')p(s'' \mid s)p(|\tau| \mid s), \tag{10}$$

where s' and s'' are the preceding and following atomic action labels in the adjacent intervals of τ. If either of them is absent, the corresponding probability is then set to be 1. For each new label assignment, the parameters of the related likelihood and prior distributions should be re-estimated. To ensure the distinctness between adjacent intervals, s can not be the same labels of the adjacent intervals.

Therefore, after randomly assigning labels for the intervals as initialization, we conduct multiple sweeps, where in each sweep, we enumerate each interval and sample a new label for it based on (9). The sampling stops when the labeling does not change after the last sweep (convergence). In practice, the sampling can converge within 100 sweeps coupled with a simulated annealing.

B. Joint Sub-Task Parsing

The joint sub-task parsing is achieved using a similar approach as atomic action parsing. We first propose the time
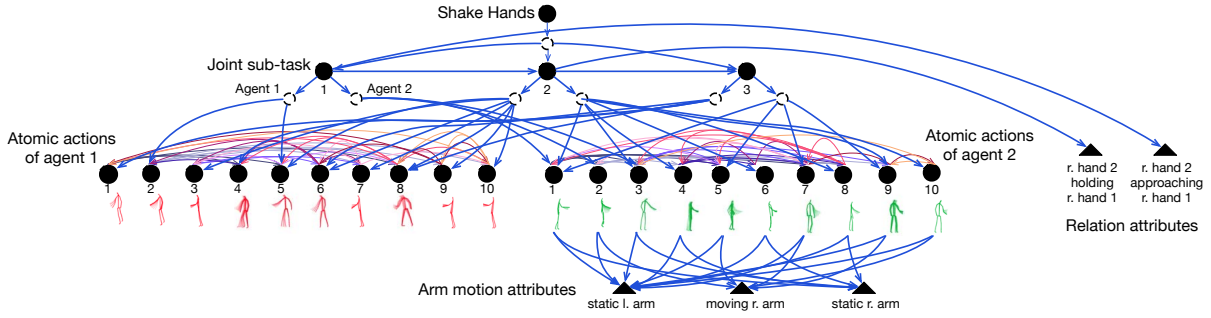
Fig. 6. The learned ST-AOG for the *Shake Hands* interaction (the motion grounding is not drawn in this figure due to the space limit). The numbers under AND nodes are the labels of joint sub-tasks or atomic actions. The edges between the atomic actions show the "followed by" temporal relations and their colors indicate which atomic actions are the edges' starting point. Similarly, the joint sub-tasks are also connected by edges representing the temporal dependencies between them. There is an example of each atomic actions from our training data, where the skeletons are overlaid with colors from light to dark to reflect the temporal order. The attributes that are not bundled to any atomic action or joint sub-task are not shown here.

intervals by detecting turning points based on the normalized sequences of O, X, and D. Then the labeling can also be optimized by a Gibbs sampling, where at each iteration, we sample a new joint sub-task label j for an interval τ by

$$j \sim p(\Gamma^R \mid \tau, j) p(\tau, j \mid \mathcal{T}^J \backslash \tau, S^J \backslash \{j^t\}_{t \in \tau}), \quad (11)$$

where $p(\Gamma^R \mid \tau, j)$ is defined in (6) and the prior probability is derived from (8) as

$$\begin{aligned} &p(\tau, j \mid \mathcal{T}^J \backslash \tau, S^J \backslash \{j^t\}_{t \in \tau}) \\ &= p(j \mid j') p(j'' \mid j) p(|\tau| \mid j) \prod_{t \in \tau} p(s_1^t \mid j) p(s_2^t \mid j). \end{aligned} \quad (12)$$

Similar to (9), j' and j'' in the above prior probability are the preceding and following intervals' joint sub-task labels. The corresponding transition probability is assumed to be 1 if either of the adjacent interval does not exist. We also constrain j to be different from the j' and j'' if they exist.

C. Constructing ST-AOG

After the previous two Gibbs sampling processes, the parameters of our probabilistic model are all estimated based on the parse graph sequences $\{PG_n\}_{n=1,\cdots,N}$. The ST-AOG of category c is then constructed by the following three steps:

Initialization. We start form a "complete" graph, where each non-leaf node is connected to all related lower level nodes (e.g., all joint sub-tasks, all atomic actions of the corresponding agent, etc.), except attribute leaf nodes.

Edge removal. Any edge between two joint sub-task nodes or two atomic action nodes is removed if it has a transition probability lower than a threshold (0.05). For each joint sub-task node, remove the edges connecting the OR node of agent i to the atomic actions whose concurrency priors under the joint sub-task are lower than 0.1. Note that we use these thresholds for all interactions.

Attributes bundling. Motion attributes: For each type of atomic action s of agent i, a *moving* attribute is bundled to a limb if the mean of the corresponding hand movement distribution specified in (2) is lower than a threshold (we use 0.2 m in practice); otherwise, a *static* attribute is bundled to the limb instead. Relation attributes: A type of joint sub-task will be associated with a *holding* attribute between a pair of hands (or a hand and an object) if the mean final hand

distance is lower than 0.15 m and the mean hand distance's change is lower than 0.1 m according to the corresponding distributions in (6). If only the mean final hand distance meets the standard, an *approaching* will be attached. For the case of multiple qualifying pairs for a hand, the one with the shortest mean distance is selected.

Fig. 6 is a learned ST-AOG for *Shake Hands* interactions. It can be seen that our learning algorithm indeed mines the critical elements of the interactions and clearly represents their relations through the structure of the ST-AOG.

VII. REAL-TIME MOTION INFERENCE

If we replace agent 2 with a humanoid, we can therefore design a real-time motion inference enabling human-robot interaction based on the learned ST-AOG by sampling parse graphs and controlling the robot's motion accordingly.

For this, we propose two levels of inference procedures: 1) robot motion generation given the parse graphs, which is essentially transferring the socially appropriate motion from agent 2 in the grammar model to a humanoid; 2) parse graph sampling given the observation of the human agent's actions and the relation between the human agent and the robot according to the learned social affordance grammar.

A. Robot Motion Generation

As shown in Fig. 3, we may use the motion grounding of agent 2 for the robot by joint mapping. The robot motion can be generated by sampling agent 2's base position $\boldsymbol{x}^t$, facing direction (i.e., base orientation of the robot) o^t, joint angles $\{\theta_{2l}\}_{l=1,2}$, and hand positions (i.e., end effector positions) $\{\boldsymbol{h}_{2l}^t\}_{l=1,2}$ at each time t based on the motion history of agent 2, $\Gamma_2^A(t-1)$, and the spatiotemporal relations, $\Gamma^R(t-1)$, upon $t-1$ as well as the agent 1's motion, $\Gamma_1^A(t)$, and parse graphs, $PG(t) = \{pg^\tau\}_{\tau=1,\cdots,t}$, upon t.

Since the arm motion is relative to the base position in our motion grounding, we first sample $\boldsymbol{x}^t$ and o^t w.r.t. the relative position and facing direction likelihood in (6), the likelihood probabilities of which must be higher than a threshold (0.05 for $\boldsymbol{x}^t$ and 0.3 for o^t). To avoid jitter, we remain the previous base position and rotation if they still meet the criteria at t.

Then we update the joint angles for each robot arm. Without the loss of generality, let us consider a single arm

Algorithm 1 Parse Graph Sampling

Input: The initial motion of two agents in $[1, T_0]$, i.e., $\Gamma(T_0)$
1: Infer $PG(T_0)$ by maximizing the posterior probability in (1)
2: Let $t \leftarrow T_0 + 1$
3: **repeat**
4:      $\Gamma' = \Gamma(t-1) \cup \{\theta_{1l}^t\}_{l=1,2} \cup \{h_{1l}^t\}_{l=1,2}$
5:      Infer current atomic action of agent 1 by
        $s_1^t = \arg\max_s p(PG(t-1) \cup \{s\} \mid \Gamma')$
6:      **for all** $j^t \in \mathcal{J}, s_2^t \in \mathcal{S}_2^c$ that are compatible with s_1^t **do**
7:         $pg^t \leftarrow \langle j^t, s_1^t, s_2^t \rangle$
8:         $PG(t) \leftarrow PG(t-1) \cup \{pg^t\}$
9:         Sample a new robot status at t, i.e., $\boldsymbol{x}^t$, o^t, $\{\theta_{2l}^t\}$ and $\{h_{2l}^t\}$, as introduced in Sec. VII-A
10:         $\Gamma(t) \leftarrow \Gamma(t-1) \cup \{\theta_{il}^t, h_{il}^t\}_{i,l=1,2} \cup \{\boldsymbol{x}^t\} \cup \{o^t\}$
11:         Compute the posterior probability $p(PG(t) \mid \Gamma(t))$
12:      **end for**
13:      Choose the pg^t and the corresponding new robot status that yield highest posterior probability to execute and update $PG(t)$ and $\Gamma(t)$ accordingly
14:      $t \leftarrow t + 1$
15: **until** $t > T$

$l \in \{1, 2\}$. According to the atomic action s_2^t, we may sample desired joint angles $\hat{\theta}_{2l}^t$ and hand position $\hat{\boldsymbol{h}}_{2l}^t$ w.r.t the corresponding likelihood terms in (2). Since we do not model the wrist orientations, the desired $\hat{w}_0$, $\hat{w}_1$, $\hat{w}_2$ are always set to be 0 if the robot arm has these degrees of freedom (Fig. 3a). If current joint sub-task entails an "approaching" or "holding" attribute for this limb, the desired hand position is set to the position of the target hand or object indicated by the attribute instead. To enforce the mechanical limits and collision avoidance, we minimize a loss function to compute the final joint angels θ_{il}^t for the robot arm:

$$\min_{\theta \in \Omega_\theta} \omega_h \underbrace{||\boldsymbol{f}_l(\theta) - \hat{\boldsymbol{h}}_{2l}^t||_2^2}_{\text{hand position loss}} + \omega_a \underbrace{||\theta - \hat{\theta}_{2l}^t||_2^2}_{\text{joint angle loss}} + \omega_s \underbrace{||\theta - \theta_{il}^{t-1}||_2^2}_{\text{smoothness loss}},$$
(13)

where $\boldsymbol{f}_l(\theta)$ is the end effector position of θ based on the forward kinematics of the robot arm l; Ω_θ is the joint angle space that follows the mechanical design (angle ranges and speed limits of arm joints) and the collision avoidance constraints, and ω_h, ω_a, ω_s are weights for the three types of loss respectively. By assigning different weights, we can design three control modes that are directly related to the attributes in ST-AOG:

1) **Hand moving mode**: if "approaching" or "holding" attributes are present in the current joint sub-task, we may use a larger ω_h to ensure an accurate hand position;

2) **Static mode**: if the first case does not hold and the atomic action has a "static" attribute for the limb, then ω_s should be much larger than ω_h and ω_a;

3) **Motion mimicking mode**: if none of the above two cases hold, we emphasize on joint angle loss (i.e., a large ω_a) to mimic the human arm motion.

In practice, we set the large weight to be 1 and the other two may range from 0 to 0.1.

B. Parse Graph Sampling

The Parse graph sampling algorithm is sketched in Alg. 1. The basic idea is to first recognize the action of agent 1. Then

following the ST-AOG, we may enumerate all possible joint sub-tasks and atomic actions of agent 2 that are compatible with agent 1's atomic action, and sample a new robot status for each of them. Finally, we choose the one with the highest posterior probability to execute. Note that the facing direction of an agent is approximated by his or her moving direction (if not static) or the pointing direction of feet (if static).

VIII. EXPERIMENTS

Dataset. There are two existing RGB-D video datasets for human-human interactions [29], [12], where the instances within the same category are very similar. To enrich the activities, we collected and compiled a new RGB-D video dataset on top of [12] using Kinect v2 as summarized in Table I, where *Wave Hands* is a new category and the instances in scenario 1 of the other categories are from [12]. For *Pull Up*, the first 3 scenarios are: A2 (agent 2) stands while A1 (agent 1) is sitting 1) on the floor or 2) in a chair; 3) A1 sits in a chair and A2 approaches. For the other categories, the four scenarios stand for: 1) both stand; 2) A1 stands and A2 approaches; 3) A1 sits and A2 stands nearby; 4) A1 sits and A2 approaches. In the experiments, we only use three fourths of the videos in scenario 1 (for *Wave Hands*, it is scenario 2) as training data, and the remaining instances are used for testing. We plan to release the dataset.

Baselines. We compare our approach with two baselines adopted from related methods, extending these method further to handle our problem. The first one (B1) uses the method proposed in [12] to synthesize human skeletons to interact with the given human agent, from which we compute the desired base positions, joint angles and hand positions for the optimization method defined in (13). Since [12] only models the end positions of the limbs explicitly and do
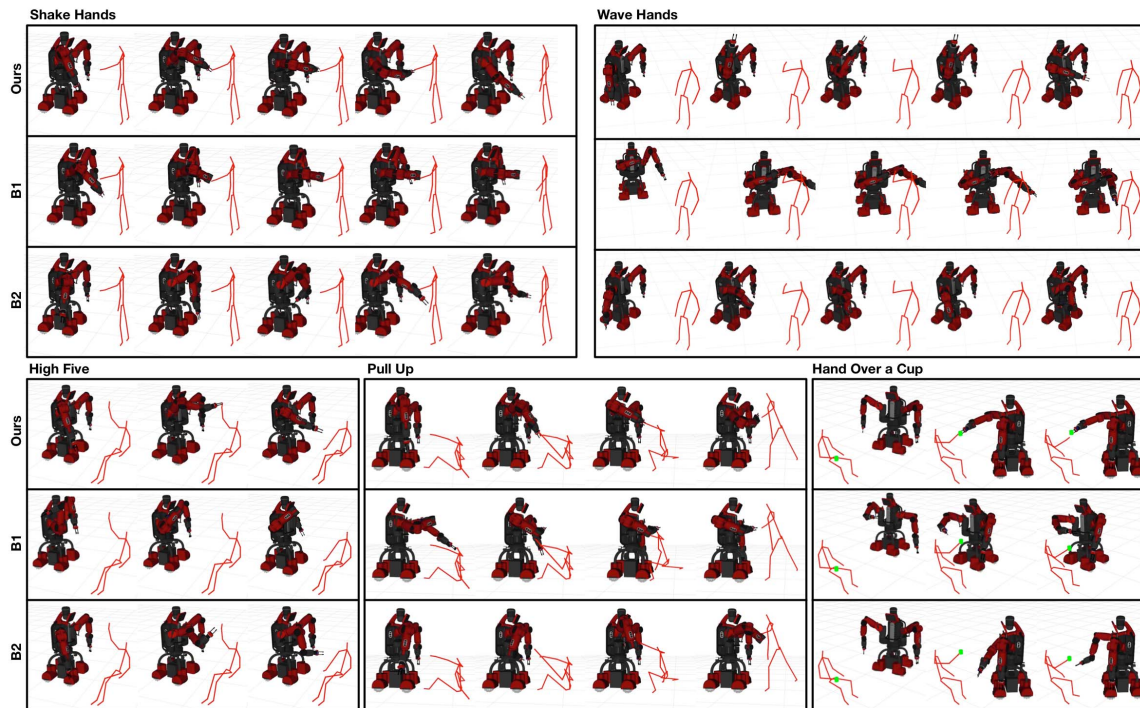
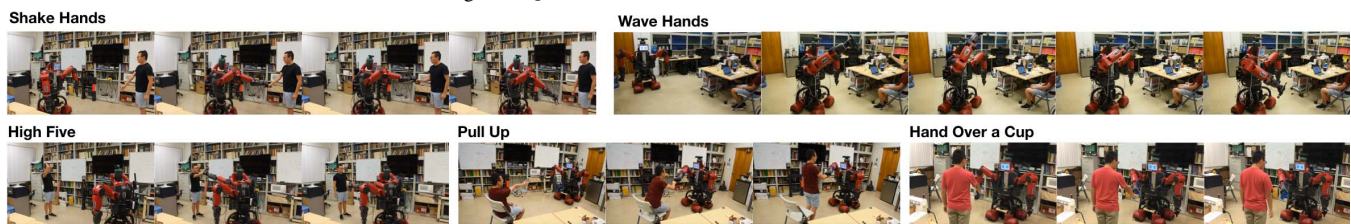Fig. 7. Qualitative results of our Baxter simulation.

Fig. 8. Qualitative results of the real Baxter test.

not specify multiple modes as ours do, we use it with the weights of hand moving mode. The second baseline (B2) uses our base positions and orientations but solve the inverse kinematics for the two arms using an off-the-shelf planner, i.e., RRT-connect [30] in MoveIt! based on the desired hand positions from our approach.

A. Experiment 1: Baxter Simulation

We first implement a Baxter simulation and compare the simulated robot behaviors generated from ours and the two baselines. For each testing instance, we give the first two frames of skeletons of two agents as the initialization; we then update the human skeleton and infer the new robot status accordingly at a rate of 5 fps in real-time. For *Hand Over*, we assume that the cup will stay in the human agent's hand unless the robot hand is close to the center of the cup (< 10 cm) for at least 0.4 s. Note that the planner in B2 is extremely slow (it may take more than 10 s to obtain a new plan), so we compute B2's simulations in an offline fashion and visualize them at 5 fps. Ours and B1 can be run in real-time.

Fig. 7 shows a simulation example for each interaction. More results are included in the video attachment. From the simulation results, we can see that the robot behaviors (standing positions, facing directions and arm gestures) generated

by ours are more realistic than the ones from baselines. Also, thanks to the learned social grammar, the robot can adapt itself to unseen situations. E.g., human agents are standing in the training data for "High Five", but the robot can still perform the interaction well when the human agent is sitting.

We also compare the mean joint angle difference between the robot and the ground truth (GT) human skeletons (i.e., agent 2) captured from Kinect as reported in Table II, which is one of the two common metrics of motion similarity [31] (the other one, i.e., comparing the end-effector positions, is not suitable in our case since humans and robots have different arm lengths). Although the robot has a different structure than humans', ours can still generate arm gestures that are significantly closer to the GT skeletons than the ones by baselines are.

B. Experiment 2: Human Evaluation

To evaluate the quality of our human-robot interactions, we showed the simulation videos of three methods to 12 human subjects (UCLA students) who did not know that videos were from different methods. Subjects first watched two RGB videos of human interactions per category. Then for each testing instance, we randomly selected one method's simulation to a subject. The subjects only watched the assigned videos

once and rated them based on two criteria: i) whether the purpose of the interaction is achieved (Q1), and ii) whether the robot's behavior looks natural (Q2). The ratings range from 1 (total failure/awkward) to 5 (successful/human-like).

The mean ratings and the standard deviations are summarized in Table III. Our approach outperforms the baselines for both criteria and has smaller standard deviations, which manifests its advantages on accurately achieving critical latent goals (e.g., holding hands) while keeping human-like motion. The rigid representation and failing to learn explicit hand relations affect B1's ability to adapt the robot to various scenarios. It also appears that only using a simple IK (B2) is probably insufficient: its optimization is only based on the current target position, which often generate a very long path and may lead to an awkward gesture. This makes the future target positions hard to reach as the target (e.g., a human hand) is constantly moving.

C. Experiment 3: Real Baxter Test

We test our approach on a Baxter research robot with a mobility base. A Kinect sensor is mounted on the top of the Baxter's head to detect and track human skeletons. To compensate the noise from Kinect, we further take advantage of the pressure sensors on the ReFlex TakkTile Hand (our Baxter's right hand) to detect holding relations between the agents' hands. Although the arm movement is notably slower than the simulation due to the mechanical limits, the interactions are generally successful and reasonably natural.

Since we only need joints on the upper body, the estimation of which is relatively reliable, the noisy Kinect skeletons usually do not greatly affect the control. In practice, temporal smoothing of the skeleton sequences is also helpful.

IX. CONCLUSIONS

We propose a general framework of learning social affordance grammar as a ST-AOG from human interaction videos and transferring such knowledge to human-robot interactions in unseen scenarios by a real-time motion inference based on the learned grammar. The experimental results demonstrate the effectiveness of our approach and its advantages over baselines. In the future, it is possible to integrate a language model into the system to achieve verbal communications between robots and humans. In addition, human intention inference can also be added to the system.

ACKNOWLEDGMENT

REFERENCES

[1] J. J. Gibson, *The ecological approach to visual perception*. Boston: Houghton Mifflin, 1979.

[2] L. Montesano, M. Lopes, A. Bernardino, and J. Santos-Victor, "Learning object affordances: From sensory-motor coordination to imitation," *IEEE Transactions on Robotics*, vol. 24, no. 1, pp. 15–26, 2008.

[3] H. Kjellström, J. Romero, and D. Kragic, "Visual object-action recognition: Inferring object affordances from human demonstration," *Computer Vision and Image Understanding*, vol. 115, no. 1, pp. 81–90, 2011.

[4] B. Moldovan, P. Moreno, M. van Otterlo, J. Santos-Victor, and L. D. Raedt, "Learning relational affordance models for robots in multi-object manipulation tasks," in *ICRA*, 2012.

[5] Y. Zhu, A. Fathi, and L. Fei-Fei, "Reasoning about object affordances in a knowledge base representation," in *ECCV*, 2014.

[6] H. Koppula and A. Saxena, "Physically-grounded spatio-temporal object affordances," in *ECCV*, 2014.

[7] A. Pieropan, C. H. Ek, and H. Kjellström, "Recognizing object affordances in terms of spatio-temporal object-object relationships," in *Humanoids*, 2014.

[8] W. Sheng, J. Du, Q. Cheng, G. Li, C. Zhu, M. Liu, and G. Xu, "Robot semantic mapping through human activity recognition: A wearable sensing and computing approach," *Robotics and Autonomous Systems*, vol. 68, pp. 47–58, 2015.

[9] Y. Zhu, Y. Zhao, and S.-C. Zhu, "Understanding tools: Task-oriented object modeling, learning and recognition," in *CVPR*, 2015.

[10] A. Gupta, S. Satkin, A. A. Efros, and M. Hebert, "From 3d scene geometry to human workspace," in *CVPR*, 2011.

[11] Y. Jiang, H. Koppula, and A. Saxena, "Hallucinated humans as the hidden context for labeling 3d scenes," in *CVPR*, 2013.

[12] T. Shu, M. S. Ryoo, and S.-C. Zhu, "Learning social affordance for human-robot interaction," in *IJCAI*, 2016.

[13] A. Gupta, P. Srinivasan, J. Shi, and L. S. Davis, "Understanding videos, constructing plots learning a visually grounded storyline model from annotated videos," in *CVPR*, 2009.

[14] W. Brendel and S. Todorovic, "Learning spatiotemporal graphs of human activities," in *ICCV*, 2011.

[15] M. Pei, Z. Si, B. Yao, and S.-C. Zhu, "Video event parsing and learning with goal and intent prediction," *Computer Vision and Image Understanding*, vol. 117, no. 10, pp. 1369–1383, 2013.

[16] T. Lan, Y. Zhu, A. R. Zamir, and S. Savarese, "Action recognition by hierarchical mid-level action elements," in *ICCV*, 2015.

[17] M. S. Ryoo and J. K. Aggarwal, "Stochastic representation and recognition of high-level group activities," *International Journal of Computer Vision*, vol. 93, no. 2, pp. 183–200, 2011.

[18] T. Lan, Y. Wang, W. Yang, S. N. Robinovitch, and G. Mori, "Discriminative latent models for recognizing contextual group activities," *IEEE TPAMI*, vol. 34, no. 8, pp. 1549–1562, 2012.

[19] M. R. Amer, D. Xie, M. Zhao, S. Todorovic, and S.-C. Zhu, "Cost-sensitive top-down/bottom-up inference for multiscale activity recognition," in *ECCV*, 2012.

[20] W. Choi and S. Savarese, "Understanding collective activities of people from videos," *IEEE TPAMI*, vol. 36, no. 6, pp. 1242–1257, 2014.

[21] T. Lan, T.-C. Chen, and S. Savarese, "A hierarchical representation for future action prediction," in *ECCV*, 2014.

[22] T. Shu, D. Xie, B. Rothrock, S. Todorovic, and S.-C. Zhu, "Joint inference of groups, events and human roles in aerial videos," in *CVPR*, 2015.

[23] Z. Deng, A. Vahdat, H. Hu, and G. Mori, "Structure inference machines: Recurrent neural networks for analyzing relations in group activity recognition," in *CVPR*, 2016.

[24] K. Lee, Y. Su, T.-K. Kim, and Y. Demiris, "A syntactic approach to robot imitation learning using probabilistic activity grammars," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1323–1334, 2013.

[25] Y. Yang, Y. Li, C. Fermuller, and Y. Aloimonos, "Robot learning manipulation action plans by "watching" unconstrained videos from the world wide web," in *AAAI*, 2015.

[26] C. Xiong, N. Shukla, W. Xiong, and S.-C. Zhu, "Robot learning with a spatial, temporal, and causal and-or graph," in *ICRA*, 2016.

[27] M. Luber, L. Spinello, J. Silva, and K. O. Arras, "Socially-aware robot navigation: A learning approach socially-aware robot navigation: A learning approach socially-aware robot navigaion: A learning approach," in *IROS*, 2012.

[28] B. Okal and K. O. Arras, "Learning socially normative robot navigation behaviors with bayesian inverse reinforcement learning," in *ICRA*, 2016.

[29] K. Yun, J. Honorio, D. Chattopadhyay, T. L. Berg, and D. Samaras, "Two-person interaction detection using body-pose features and multiple instance learning," in *CVPR Workshops*, 2012.

[30] J. J. Kuffner and S. M. LaValle, "Rrt-connect: An efficient approach to single-query path planning," in *ICRA*, 2000.

[31] M. Alibeigi, S. Rabiee, and M. N. Ahmadabadi, "Inverse kinematics based human mimicking system using skeletal tracking technology," *JIRS*, vol. 85, pp. 27–45, 2017.