A Framework for User-Defined Body Gestures to Control a Humanoid Robot

Mohammad Obaid · Felix Kistler · Markus Häring · René Bühling · Elisabeth André

Abstract This paper presents a framework that allows users to interact with and navigate a humanoid robot using body gestures. The first part of the paper describes a study to define intuitive gestures for eleven navigational commands based on analyzing 385 gestures performed by 35 participants. From the study results, we present a taxonomy of the user-defined gesture sets, agreement scores for the gesture sets, and time performances of the gesture motions. The second part of the paper presents a full body interaction system for recognizing the user-defined gestures. We evaluate the system by recruiting 22 participants to test for the accuracy of the proposed system. The results show that most of the defined gestures can be successfully recognized with a precision between 86−100 % and an accuracy between 73−96 %. We discuss the limitations of the system and present future work improvements.

Keywords Humanoid robot · Robot · Nao · Gesture · User-defined · User-defined gestures · Robot navigation · Gesture recognition

1 Introduction

Markerless body tracking technologies based on depth sensors allowed researchers to have an easy-to-use platform for developing algorithms for recognizing full body gestures and postures in real time [1,2]. Recently, researchers are increasingly addressing the use of algorithms to recognize full body gestures and postures, in real time, to teleoperate and guide robots and hence enhance the user's natural experience and engagement with the robot, such as the work by [3,4]. The key to their approaches is to define intuitive and natural human-robot interaction (HRI) using non-verbal communications, such as body gestures. Generally, most of the algorithms that use body gestures to control robots are based on gesture design paradigms that are defined by their developers. However, as the user is not involved in the process, the designed gestures may not be the most intuitive and may not represent their natural behavior. More recently, several researchers have addressed the same problem with the design of gesture based interaction methods in several other domains including surface computing [5] and public displays [6]. However, a user-defined set of gestures for the control of a humanoid robot has not been defined to this date. In order to support the control of robots using true natural full body interaction, we need to collect data on the basis of the users body behavior.

We present the design of a framework based on the users natural behavior to navigate a humanoid robot. We collect data from both Technical[1] (T) and Non-Technical (NT) users when performing gesture motions to navigate a humanoid robot (Nao by Aldebaran Robotics[2]). We contribute to the field of HRI the following: (1) the establishment of user-defined gesture sets for both (T and NT users) to navigate a humanoid robot, (2) the analysis of qualitative and quantitative data that includes gesture taxonomy, performance data measures, observations, and subjective responses, (3) an understanding of the implications for humanoid robot control

M. Obaid (✉)
t2i Lab, Chalmers University of Technology, Gothenburg, Sweden
e-mail: mobaid@chalmers.se

F. Kistler · M. Häring · R. Bühling · E. André
Human Centered Multimedia, Augsburg University, Augsburg, Germany

[1] We term a user that is experienced with robots and/or gesture tracking as *Technical*

[2] http://www.aldebaran-robotics.com.

using human gestures, and (4) the development of a system that can be used to recognize the user-defined gestures to navigate a robot.

2 Background

Alongside verbal communications, the non-verbal communication channels in the human-humanoid robot interaction have been recognized by researchers as one of the important aspects to interact with a robot in an intuitive and a flaunt way [7–11]. Moreover, scientists recently presented the use of human body gestures as one of the main non-verbal communicative cues to serve as a tool in signalling actions to social robots. The following sections explore related literature on human gestures, designing gestures, and gesture controlled robots.

2.1 Human Gesture Categories

Kinesics is the study of human gestures and body postures in the field of non-verbal communication behaviors. Researchers have conducted a vast number of studies to understand gestural interactions between individuals and how gestures can be categorized based on the information communicated. There is no universal categorization standard for body gestures and postures, however, researchers used different taxonomies for categorization. Efron [12] was one of the first to classify gestures into five categories: physiographics, kinetographics, ideographics, deictics, and batons. While, Ekman and Friesen [13] specified four categories of the human gestures and postures based on the communication function: emblems, illustrators, regulators and adaptors. McNeill [14] presented five types of gestures: cohesive, beat, deictic, iconic, and metaphoric gestures. Beat gestures are rhythmic movements that point out particular parts of a speech, while cohesive try to keep up the continuity. Deictic gestures can be defined as a pointing reference in space. Iconic gestures or illustrators are gestures that relate to speech to help describe the speech content. The metaphoric gestures shape abstract concepts to explain an idea. Moreover, McNeil [15] defined four phases that construct a gesture: preparation, stroke, hold, and retraction. The preparation is the phase that brings the body from its rest to a position that is suitable for executing the gesture. The stroke phase is the real information contained in the gesture, while the retraction is the phase where the body goes to its rest position again. Some gestures, especially pointing gestures, also have an extended hold phase after the actual stroke in which the arms remain in their position for a while. In this paper, we use the phases defined by McNeil as we found his work to have a well defined construct of gesture phases and a well defined annotation practice scheme [16].

2.2 Designing Gestural Input

The basic rule when designing an interface is to initially define the needs of its users and gestural interfaces are no exception [17]. Therefore, several domain areas employ the design of appropriate gestures for a system by allowing users to intuitively define how they would use it. Recently, the work presented by Wobbrock et al. [5] described the design of appropriate gestures for surface tabletop interfaces. They define gestures by employing NT users to observe the effect of a gesture and then asked them to perform a gesture to match its cause. The work by Wobbrock et al. was a motive for many researchers to follow a similar design paradigm in their fields. For example, Ruiz et al. [18] presented results of a user-defined motion gesture set for smartphone interactions. Kray et al. [19] identified user-defined gestures that can be used to communicate a mobile phone with public display, tabletops, and other devices. Their results also revealed which phone sensors can be used to achieve a better recognition of common user-defined gestures. Kurdyukova et al. [6] presented a study for identifying a user-defined set to transfer data using an iPad in a multi-display environment. Kistler et al. [1] investigated user-defined full body gestures for an interactive storytelling scenario. They identified gestures that are an intuitive representation for a specific set of in-game actions. These in-game actions triggered navigation and dialogs within the virtual scenario. In this research, we follow a similar approach to Wobbrock et al., with a focus on gestural controls for humanoid robots.

2.3 Gesture Controlled Robots

In general, navigational control of a humanoid robot are done using traditional input computer devices, such as a keyboard and mouse [20,21] or a joystick [22]. However, the fact that humanoid robots are machines that look like humans and preserve some human functionalities has motivated researchers to look for intuitive interaction ways that are similar to the human-human communications. Thus, several researchers looked at how to interpret and classify the human body poses and gestures to improve the HRI, such as the work presented in [23–28]. Among the efforts for intuitive gestural communications in HRI is controlling humanoid robots using human gestures and a natural input method. Waldherr et al. [29] presented a gesture based robot control interface to ease interaction with a mobile robot. Their work instructs a robot to follow gestural commands to clean-up an office. They used a vision-based approach to detect users and recognise their gestures (both static and motion gestures). They predefined multiple prototypes for each gesture (stop, follow, pointing vertical and pointing low) and trained their system to recognise them. Nhan Nguyen-Duc-Thanh et al. [30] demonstrates a new approach to control a humanoid robot

(Nao) using human body language. Their method is based on a Semaphore alphabetical system, where the human body poses and gestures are recognized, with the help of Kinect sensors, as alphabetical characters that can be interpreted by the robot. Broccia et al. [31] use Kinect to recognize the users upper body movements, which are mimicked by a humanoid robot based on a mathematical mapping of the human movements to the robots joints. For navigational purposes they use full body gestures like stepping forward, stepping backward and turning the body. On the other hand, Cabibihan et al. [32], conducted a study to investigate the recognition of 15 human-like gestures that are performed by a human actor and an anthropomorphic robot. Their results revealed that 8 gestures were recognized from the human-actor's video and six gestures are recognized from the robot's video. Their work address what robot gestures are understood by humans, which is the opposite to defining gestures to control a robot. Strobel et al. [33] presented a system architecture to identify six human gestures to interact with a cleaning assistant robot in a domestic environment. Their focus is to recognise the users intent behind six predefined gestural actions by training a hidden Markov model with a set of gestural examples. Moreover, Stiefelhagen et al. [3] employed multimodal approaches by combining speech with gesture commands, while other work efforts are put towards controlling robots using pointing gestures [34,35], but such methods are limited to a certain range of commands. Hu et al. [36] developed simple hand gestures for robot navigational actions, while, the recent work of Konda et al. [37] employ full body postures to navigate their robots. However, none of the above have considered the design of user-defined gestures to control a robot and produce an intuitive gesture set based on the users' preferences, which is the core part of the work presented in this paper.

Previous work, in this field, relied on the developers of the system to define commands and gestural instructions while approaches that follow a user-centered design approach are rare. An example includes the work by Dillmann et al. [38] (which is in line with the overview given by Breazeal and Scassellati [39]) proposed a system to teach a humanoid robot assistive tasks through observing a human user. Barattini et al. [40] who defined a gesture set for the control of industrial collaborative robots based on user-centered design criteria, such as physical and mental effort. Ende et al. [41] as well as Gleeson et al. [42] defined gesture sets for robot control based on observations of human-human collaboration. The underlying assumption is that gestures inspired by human-human interaction are easy to remember and to perform. An approach to gesture design similar to our own approach has been presented by Bodiroa et al. [43]. They conducted an experiment in which they asked users to perform gestures they associated with a given task that was described with verb-noun keywords, such as "bring check". While the approach served to identify appropriate gestures for human-robot control, the resulting gestures have not yet been evaluated in such a scenario. Our approach distinguishes from their work by presenting users with videos of robots performing a task as opposed to describing the task verbally. The advantage of our experimental setting for acquiring gestures is the greater similarity that it bears to the setting in which the gestures will be eventually employed.

3 User Study: Defined Gestures to Control Humanoid Robots

The main objective of this study is to define a set of control body gestures derived from the users actions when intuitively instructing a humanoid robot. In particular, in this study, we focus on navigational control of the humanoid robot Nao. We use eleven actions (*Move forward, Move backward, Move left, Move right, Turn left, Turn right, Stop movement, Speed up, Slow down, Stand up, Sit down*) for which users, of the presented study, chose gestures.

The motions of all navigational actions are implemented from the perspective of the robot using the built in motion module of the Nao system (Academic Edition V3.2). We teleoperate the robot through a WiFi connection by implementing several python scripts that use the native API delivered by Aldebaran Robotics. We adopt the Wizard-of-Oz technique to teleoperate the robot throughout each session.

In this section we describe a study to identify a set of common intuitive gestures to control a humanoid robot. Additionally, we want to see whether users with a technical background with a better understanding of gesture recognition hardware, like Kinect, and knowledge about robots and their abilities, use different gestures than participants without a technical background.

3.1 Participants

To define a set of intuitive gestures to control a humanoid robot, we consider two types of user groups, Technical (T) and Non-Technical (NT): The first are users that have some experience with humanoid robots and are aware of gesture tracking systems (such as Microsoft Kinect). The second are users that do not have much experience with such technologies. We consider the two groups as it is apparent when a user is aware of the limitation of the technologies they can define their gestures based on those limitations; hence, including the two groups (T and NT) allows system designers to consider the characteristics of both groups.

We elicit performed gestural actions from 35 participants (17 T, 18 NT), all from Germany. Initially, we asked participants, on a 5-point Likert scale (ranging from one to five), about their experience with the Microsoft Kinect and with a

Table 1 Taxonomy for full body gestures used to control a humanoid robot based on 385 gestures

Taxonomy of full body gestures for controlling a humanoid robot		
Form	Static gesture	A static body gesture is held after a preparation phase
	Dynamic gesture	The gesture contains movement of one or more body parts during the stroke phase
Body Parts	One hand	The gesture is performed with one hand
	Two hands	The gesture is performed with two hands
	Full body	The gesture is performed with at least one other body part than the hands
View-Point	Independent	The gesture is independent from the view point
	User-centric	The gesture is performed from the user's point of view
	Robot-centric	The gesture is performed from the robot's point of view
Nature	Deictic	The gesture is indicating a position or direction
	Iconic	The gesture visually depicts an icon
	Niming	The used gesture is equal to the meant action

humanoid robot. The 17 T participants (6 female, 11 male) have an average experience with MS Kinect = 2.71 and with a humanoid robot = 2.41. The 17 T participants have an average age of 29 ($SD = 5.2$) and are mainly from the Computer Science background. While the 18 NT participants (10 female, 8 male) have an average experience with MS Kinect = 1.11 and with a humanoid robot = 1.06. Most of the 18 NT participants are students from several disciplines, such as education, languages or economics, and have an average age of 27 ($SD = 7.8$). All participants except one were right-handed.

3.2 Apparatus

The experiment is arranged in a room that is 3 meters wide and 6.5 meters deep. The room is equipped with a 50 inch plasma display and two cameras. The first camera records the front view of the user, while the other camera is setup as a side camera. The user has a designated region that he/she is allowed to freely move in during the study. This region is defined from the user's initial position and a distance of about 1 meter around that point. The humanoid robot is placed 2 meters away from the user and is facing them.

3.3 Procedure

At the beginning of the experiment, each participant is given a description of the study and are told to stay within their designated region in the room. The following are the steps each participant is asked to follow: (1) on the screen, watch a video that demonstrates how Nao performs one of the navigational actions. (2) Upon the completion of the video, perform a gesture that can command Nao to repeat the demonstrated action. (3) Watch Nao performing the corresponding action (this is remotely activated by an instructor). (4) Answer a questionnaire corresponding to the action.

The eleven actions are presented to each participant in a randomized order. For the actions *Speed up, Slow down* and *Stop movement*, Nao will be in motion when the gesture is to be preformed by the participant. In this case, participants are asked to state when they are ready, after watching the video on the screen, and Nao is immediately activated then. Subjective and objective measures are explained further in Sect. 4.

4 Results

The results of our study presents a gesture taxonomy, a user-defined gesture set, performance data measures, qualitative observations, and subjective responses.

4.1 Gesture Taxonomy

We manually classify all gestures according to four dimensions: *form*, *(involved) body parts*, *view-point*, and *nature*. Each dimension consists of multiple items, shown in Table 1. They are partly based on the Taxonomy used by Wobbrock et al. [5] and adapted to match full body gestures. Moreover, *nature* was inspired by gesture categories defined by Salem et al. [8].

Form distinguishes between static and dynamic gestures (without and with movement respectively). Static gestures have a preparation phase at the beginning, in which the user moves into the gesture space, but the core part of gesture is after the preparation phase. Therefore, the gesture is kept for a certain amount of time before the user releases it again in the retraction phase. In opposite, dynamic gestures have a clear stroke phase including the movement of specific body parts between the preparation and retraction phases.

The *body parts* dimension is self-explanatory. It distinguishes between one hand, two hands, and full body gestures that involve at least one other body part.

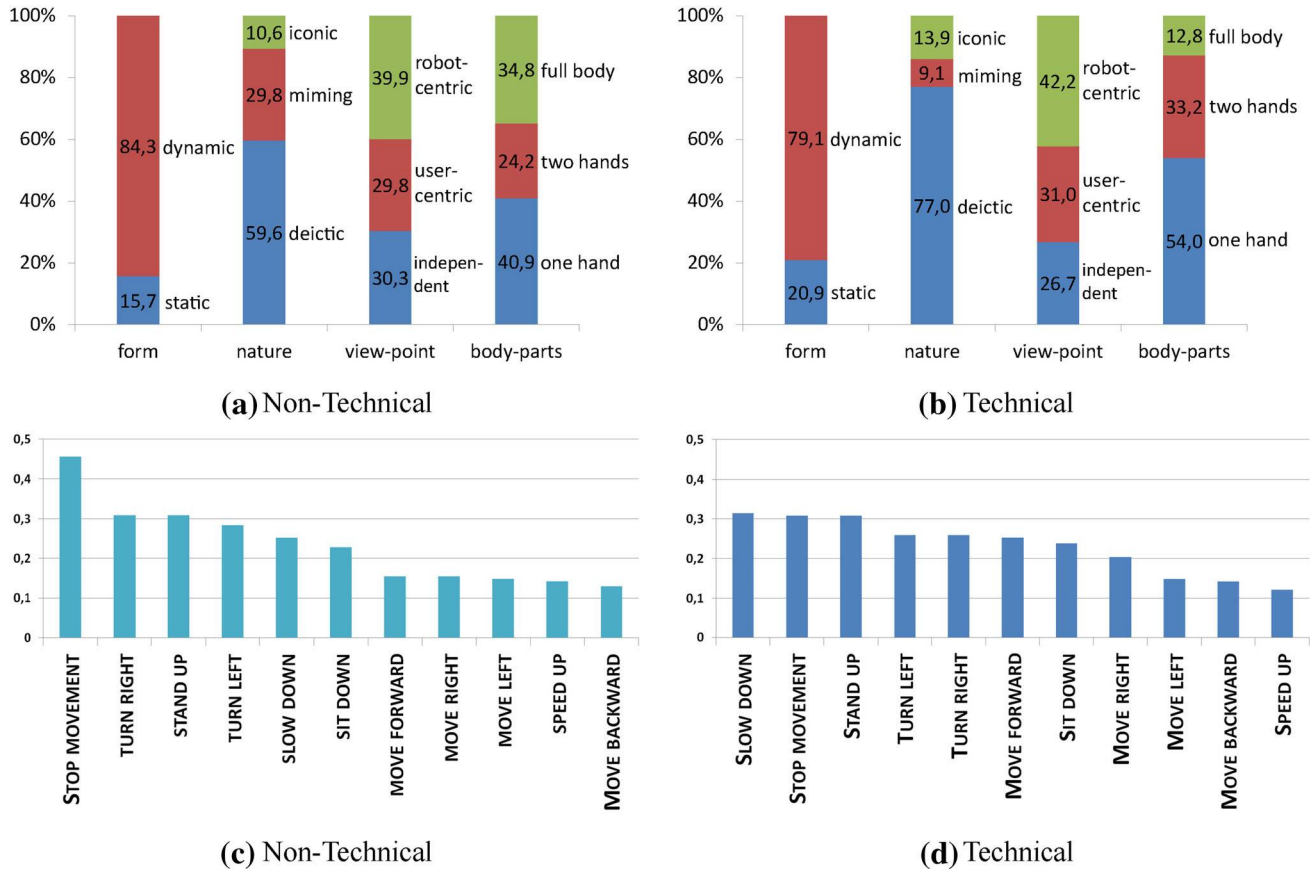


Fig. 1 Taxonomy distribution (**a**) and (**b**) and gesture agreement levels (**c**) and (**d**) for technical and non-technical users

The *view-point* dimension can be explained best with pointing gestures in a scenario where the robot is facing the user. Thus, a user-centric view-point means that when the user is pointing to his/her right, the robot should move in the pointing direction and, therefore, to the left from the robot's view. The opposite is a robot-centric view-point, i.e. when the user is pointing to his/her right, the robot moves in opposite to the pointing direction (to the right from the robot's view). Other gestures are view-point independent, for example, an open front-facing hand for stop which does not include any directional information.

The *nature* of our gesture is divided in three categories: The most common gestures we found for HRI are deictic gestures, that indicate a position or direction. These gestures can be either static, e.g. pointing to the right, or dynamic, e.g. waving to the right. They can be performed with one hand, two hands, or even other body parts, e.g. tilting the head. They can be performed from a user-centric or robot-centric view-point. Iconic gestures are visual depictions, e.g. an open front-facing hand for stop, or drawing a circle in the air for turning. Miming gestures realize the idea that the user shows the robot how to perform the action by actually performing it, e.g. if the action is sitting down, the user actually sits down. Depending on the view-point, miming gestures can be mirrored as well.

Figure 1 depicts the taxonomy distributions for T and NT users. The two most visible differences between the two kinds of users can be seen in the *nature* dimension (χ^2 (2) = 26.36, $p < 0.001$) and the *involved body parts* dimension (χ^2 (2) = 25.46, $p < 0.001$). While T users clearly prefer deictic gestures and mainly use their hands for gesturing, NT users more often use full body and miming gestures. Therefore, one can say that T users prefer more abstract and less exhausting gestures. This is emphasized by the fact that the T users also tend to use more static postures than the NT, however, we found no significant differences for the *form* dimension (χ^2 (1) = 1.75, $p = 0.186$).

4.2 A User-Defined Gesture Set

The gestural data collected from the participants of the study, to control the Humanoid Robot Nao, is used to define a set of user-defined gestures that can be used for the specified control actions. The process of selecting a suitable gesture for a control action is as follows:

- For each control action t we identify a set P_t that contains all proposed gestures.
- The proposed gestures in P_t are then grouped into subsets of identical gestures $P_{i_{1..N}}$, where i is a subset that contains identical gestures and N is the total number identified subsets.
- The representative gesture for t is identified by selecting the subset P_i with the largest size, i.e. $MAX(P_i)$

To further evaluate the degree of agreement among participants towards the selected user-defined sets, we employ a process that computes an agreement score based on the work defined and used by Wobbrock et al. [5], [44]. An agreement score S_t corresponding to a selected user-defined gesture for action t is represented by a number in the range $[1/|P_t|, 1]$ that defines the general agreement among participants. Wobbrock et al [44] presented the following equation to calculate the agreement score:

$$S_t = \sum_{P_i} \left(\frac{|P_i|}{|P_t|} \right)^2 \quad (1)$$

The results of evaluating the degree of agreement for the eleven control actions of our study are presented in Fig. 1 (c) and (d). The overall agreement levels for the T and NT participants are the same, $S = 0.23$.

Figure 2 depicts the representing gestures for the eleven actions for both T and NT users. In some cases, two representative gestures are present for one action as there were two large size gestural subsets (P_i) with an equal number of identical gestures, e.g. Action 1 for NT.

4.3 Gestural Phases and Timing

The video recordings of all participants, from the camera videotaping the frontal view, were annotated using the ELan annotation tools.[3] The annotations segmented each video into 11 actions and each action into 4 phases (Start-up, Preparation, Stroke, and Retraction). The start-up phase represents the time it takes the participants to start their gestural instruction, after watching the action on the screen. The others are the times for the gestural phases defined by McNeill [15]. Using the annotation tool, the times for the 4 phases are extracted for the 11 actions of each participant. Figure 2 shows the average times (for T and NT) for each of the phases of each gesture representing an action.

4.4 Subjective Ratings

After each action, participants are asked to rate the *goodness* and *easiness* of their performed gesture on 7-point Likert scales. The results reveal that the *goodness* of the gestures and the *easiness* to think of them correlated significantly for the T group ($r = 0.54$, $p < 0.01$) as well as for the NT group ($r = 0.40$, $p < 0.01$). As expected, gestures that are considered as good matches for an action are usually easy to think of and produce. Beside the direct correlation between *goodness* and *easiness*, we checked for their correlation with the level of agreement and the timings (especially the *Start Up* and *Stroke* phase) but nothing significant was found.

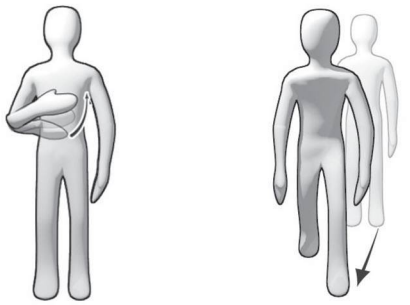
4.5 Implication for Gesture Recognition

Most user-defined gestures for navigational control of a humanoid robot are deictic gestures, which indicate a position or direction. Therefore, the main focus of the gesture recognition should lay on these type of gestures. However, we notice that the gesture view-point may vary especially in these cases. This poses a great challenge for the gesture recognition: if mirrored gestures should be allowed, how does the robot know if it should move to the left-hand or right-hand side, when the user is pointing to his/her right? A solution could be to offer different modes for the navigational control: one in robot-view and one in user-view. Nevertheless, the interaction designer should think carefully of which gestures are influenced by the control mode. For example, gestures for linear movements are usually all influenced depending on the chosen view-point, while gestures for rotating the robot remain the same. Another interesting point is, that one-hand gestures are still the most important ones, however two-hand gestures are also used quite often, and NT users also performed quite a lot of gestures that involve other body parts. The usage of the second hand mostly results in symmetrical gestures, for which the information from the second hand is, more or less, redundant, but could be used to increase the confidence of a recognition system. The use of full body gestures raises a different issue: they can only be included when implementing additional gesture recognizers, and in opposite to the hand gestures, they really need the full body tracking information which justifies the usage of a depth sensor with corresponding tracking technology. Users generally preformed dynamic gestures, therefore, simple posture recognition would often be not enough. Moreover, the usual statically labeled pointing gesture should not be optimized for a certain amount of dwell-time as a lot of users included a single or repeated waving motion into pointing to indicate direction.

In the following section, we present a system for recognizing the user-defined gestures for the purpose of controlling a humanoid robot using eleven navigational actions in real-time.

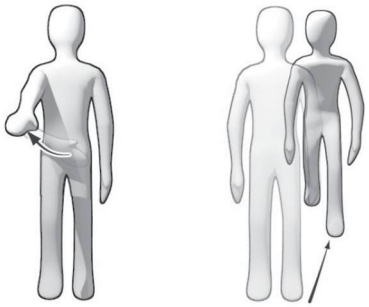
[3] Max Planck Institute for Psycholinguistics, Nijmegen, The Netherlands (http://www.lat-mpi.eu/tools/elan/).

Move forward



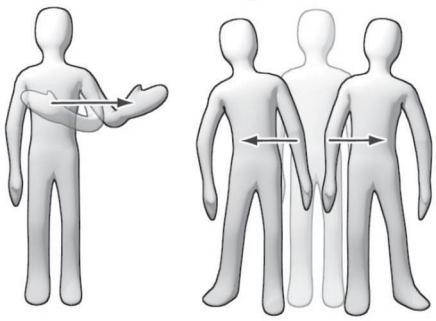
	T	NT (a)	NT (b)
Startup	2.89, 1.71	2.12, 1.52	2.04, 0.65
Preparation	0.29, 0.05	0.15, 0.30	0.54, 0.34
Stroke	1.84, 1.21	4.84, 1.87	2.02, 1.37
Retraction	0.80, 0.44	0.72, 1.44	0.63, 0.17

Move backward



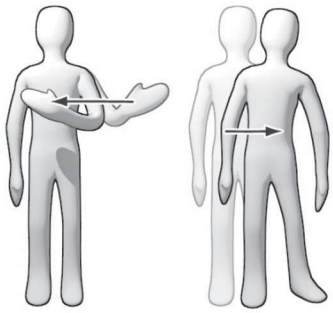
	T	NT
Startup	2.26, 0.91	2.19, 1.57
Preparation	0.40, 0.23	0.27, 0.54
Stroke	1.61, 0.47	4.22, 2.18
Retraction	1.12, 0.79	N/A

Move right



	T	NT (a)	NT (b)
Startup	3.00, 1.43	1.00, 0.52	2.39, 0.69
Preparation	0.42, 0.39	0.17, 0.33	0.24, 0.28
Stroke	2.04, 1.91	5.10, 1.65	2.85, 1.38
Retraction	0.84, 0.53	N/A	0.50, 1.01

Move left



	T	NT
Startup	1.78, 0.60	1.44, 0.62
Preparation	0.70, 0.62	0.09, 0.19
Stroke	2.37, 2.34	4.59, 1.54
Retraction	0.80, 0.24	N/A

Turn left

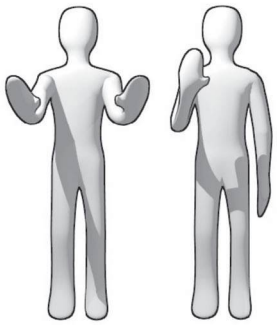
	T	NT
Startup	2.59, 1.43	2.32, 2.18
Preparation	0.59, 0.57	0.73, 0.76
Stroke	2.28, 1.41	3.91, 2.45
Retraction	1.36, 0.86	0.70, 0.69

Turn right

	T	NT
Startup	2.48, 1.10	1.10, 0.48
Preparation	0.43, 0.18	0.45, 0.21
Stroke	2.19, 0.97	2.83, 1.60
Retraction	0.75, 0.40	0.73, 0.31

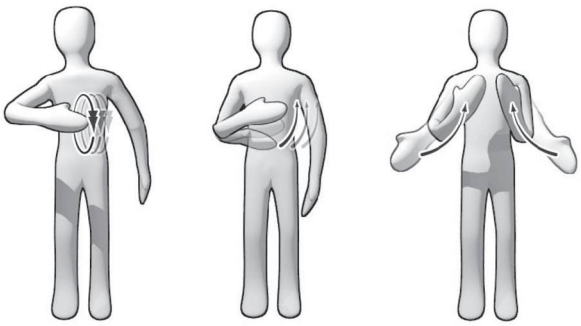
Fig. 2 User-defined gesture sets for the technical (T) and non-technical (NT) participants to navigate a humanoid robot. Values: Mean and SD in seconds

Stop movement



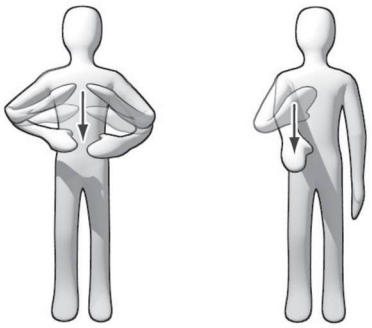
	T	NT
Startup	2.76, 1.26	2.48, 1.26
Preparation	0.32, 0.12	0.24, 0.13
Stroke	2.14, 0.46	2.52, 0.92
Retraction	0.57, 0.25	1.09, 1.09

Speed up



	T (a)	T (b)	NT
Startup	2.07, 1.54	2.27, 1.04	2.67, 1.36
Preparation	0.24, 0.07	0.25, 0.05	0.28, 0.13
Stroke	3.05, 1.51	1.60, 0.11	2.86, 2.03
Retraction	0.74, 0.44	0.63, 0.10	0.95, 0.28

Slow down

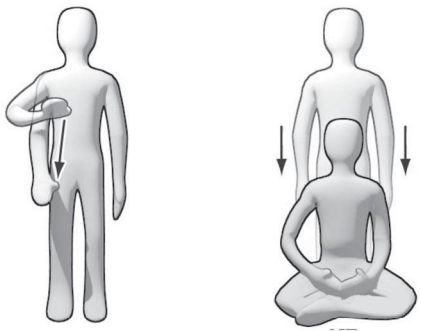


	T	NT (a)	NT (b)
Startup	3.12, 2.36	2.81, 1.17	2.92, 2.24
Preparation	0.37, 0.31	1.06, 1.46	0.89, 1.50
Stroke	3.72, 1.76	3.09, 1.23	2.56, 1.73
Retraction	1.25, 0.83	1.16, 0.47	1.35, 0.80

Stand up

	T	NT
Startup	2.71, 0.92	1.93, 0.94
Preparation	0.57, 0.38	0.72, 0.72
Stroke	2.00, 1.21	3.67, 1.92
Retraction	0.95, 0.22	0.38, 0.38

Sit down



	T	NT (a)	NT (b)
Startup	1.69, 0.65	1.35, 0.27	2.46, 1.58
Preparation	0.37, 0.18	0.35, 0.11	0.99, 0.66
Stroke	2.11, 2.29	1.81, 1.05	6.92, 2.32
Retraction	0.76, 0.21	0.58, 0.12	1.52, 2.08

Fig. 2 continued

5 A Full Body Interaction Framework

The user-defined gestures to control a humanoid robot (described in Sect. 4.2) require a recognition system that is able to classify distinct features of a performed gesture in real-time. In this section, we therefore, present a full body recognition system that can be customized to classify gestures for the purpose of controlling a humanoid robot. We present the system setup, implementations, followed by an evaluation and a discussion of the system. To demonstrate the different feature of the system and its recognition accuracy, we use a subset of the user-defined gestures presented in Sect. 4.2, in particular, we use the defined gestures by the NT users[4] (as shown in Fig. 2).

5.1 Recognition System

The recognition system was developed on the basis of our work on the Full Body Interaction framework (FUBI),[5] of which an earlier version is described in [1]. FUBI can use OpenNI/NiTE[6] or the Kinect for Windows SDK[7] for applying full body user tracking on the data of a depth sensor. For evaluating our system we decided to use the Kinect SDK with a Kinect sensor. In this way, we get positions and orientations for 20 different user joints. The joint data is analyzed in the recognition framework for detecting gestures that are defined via XML files. Those XML files first can contain four types of basic gesture recognizers:

1. Joint orientation recognizers are defined by a minimum and/ or a maximum angle for a specific joint.
2. Joint relation recognizers looks at the position of a joint in relation to another. For example, whether a joint is above other joints and how far is a joint above another joint.
3. Linear movement recognizers are defined by a specific direction and a minimum and/or a maximum speed.
4. Finger count recognizer attempts to detect a minimum and/ or maximum number of displayed fingers.

In addition, those three types of basic recognizers can be combined in a sequence to form a combination recognizer. A combination recognizer describes a deterministic linear state machine and consists of several states that contain sets of the above mentioned basic recognizers. For each state and the recognizer it references, several attributes can be set to define

```
<JointRelationRecognizer name="rightHandFront">
  <Joints main="rightHand" relative="rightShoulder"/>
  <MaxValues z="-200" />
  <MinValues y="-200"/>
</JointRelationRecognizer>
<JointRelationRecognizer name="rightHandCloseToSensor">
  <Joints main="rightHand"/>
  <MaxValues z="2000"/>
</JointRelationRecognizer>
<FingerCountRecognizer name="OpenHand">
  <Joint name="rightHand"/>
  <FingerCount min="3"/>
</FingerCountRecognizer>
<CombinationRecognizer name="RightHandFrontOpen">
  <State minDuration="0.5" >
    <Recognizer name="rightHandFront"
      ignoreOnTrackingError="true"/>
<!-- ->still works if shoulder tracking failed-->
    <Recognizer name="rightHandCloseToSensor"/>
  </State>
  <State minDuration="0.5">
    <Recognizer name="OpenHand"/>
    <Recognizer name="rightHandFront"
      ignoreOnTrackingError="true"/>
    <Recognizer name="rightHandCloseToSensor"/>
  </State>
</CombinationRecognizer>
```

Fig. 3 The XML definition to recognize the Stop movement action

the properties of a gesture. The most important attributes are a minimum and a maximum duration, which recognizers have to fulfill in the recognition process to get into and stay in the that state. Another important attribute is the maximum allowed duration for the transition to the next state. Figure 3 depicts the XML definition for a combination recognizer we implemented to recognize the gesture most NT users chose for the action Stop movement. The FUBI framework allows us to implemented the user-defined gestures in our system very quickly, and to test whether it is feasible to recognize them using the tracking data provided by OpenNI/NiTE or the Kinect for Windows SDK.

5.2 User-Defined Gesture Classification

In this section, we describe the process used to classify the user-defined gestures (from Sect. 4.2). To demonstrate, we present the implementation aspects for a test set of eleven gesture candidates of the NT users (illustrated in Fig. 2). Considering the implications for gesture recognition (Sect. 4.5), we eliminated any duplicate or similar gestures in the set; for example, there are two NT gesture candidates for the action Move forward, one deictic and one miming gesture, on the other hand, there are only miming gestures for Move backward/ left/ right. Therefore, we chose the miming gesture for the Move forward action to stay consistent with the other Move actions.

Furthermore, there are near identical NT gesture candidates for the actions Slow down (NT (b)) and Sit down (NT (a)), thus, we decided to use the NT (b) of the Sit down action.

[4] It is important to note that the NT user-defined gesture set is used to demonstrate the various parts of the recognition system and its accuracy; thus, the system is not limited to the NT gesture set only.

[5] http://hcm-lab.de/fubi.html.

[6] http://www.openni.org.

[7] http://www.kinectforwindows.org.

Table 2 Confusion Matrix for the recognition of the 11 implemented gesture candidates

Action	Gesture (cf. Fig. 2)	True Positives	False Positives	False Negatives	Precision	Accuracy	Recall
Move forward	NT (b)	42	2	0	95 %	95 %	100 %
Move backward	NT	43	2	0	96 %	96 %	100 %
Move right	NT (a)	39	0	5	100 %	89 %	89 %
Move left	NT (b)	42	1	1	98 %	95 %	98 %
Turn left	T/NT	20	4	21	83 %	44 %	49 %
Turn right	T/NT	18	3	23	86 %	41 %	44 %
Stop movement	NT	32	5	7	86 %	73 %	82 %
Speed up	NT	5	21	16	19 %	12 %	24 %
Slow down	NT (b)	43	0	2	100 %	96 %	96 %
Stand up	T/NT	39	1	3	98 %	91 %	93 %
Sit down	NT (b) modified	37	5	2	88 %	84 %	95 %
Overall		360	44	80	89 %	74 %	82 %
Overall w/o Speed up		355	25	64	93 %	80 %	85 %
Overall w/o Speed up, Turn left/right		317	18	20	95 %	89 %	94 %

The gestures we chose for all actions can be seen in the second column of Table 2. Moreover, gestures that include repeated movements within the stroke phase are implemented to be classified from a single performance of that movement.

According to the timings displayed in Fig. 2 and the video recordings of the gesture performances, we tried to implement the gesture classifications as close as possible to how the users actually performed them for each action. However, there are some restrictions of the depth sensor based tracking system that we needed to take into account as well. The most obvious modification was for the action Sit down, in which we adopted the sitting position. However, when a user is really sitting down on the ground, the tracking becomes very unstable, looses several joints, and sometimes even fails completely. Therefore, we chose to modify this motion, so that the user should not completely sit down on the ground, but only adopt to a squatting position with the knees bent about 90 degrees. The recognizer for this gesture was accordingly looking at the orientation of both knee joints, and waiting for a high rotation around the x-axis that needs to be kept for at least 0.5 seconds.

The recognizers for the actions Move forward/ backward/ right/ left, were all implemented in a similar way. At first, they all require at least a short period with no body movement to avoid multiple detections within one performance. After that, they expect a movement in the corresponding direction, which lasts long enough to be able to perform at least one step in that direction. The time constraint was adjusted according to the minimum time it took participants to perform one step during the gesture performances, which was between 0.4 and 0.7 s depending on the direction.

For the actions Turn right/ left there was only one gesture candidate, i.e. drawing a circle with one hand in the x-z-plane. The recognizers separated this movement into 6 parts, so the Turn right recognizer waited for the hand sequence movements directed right-forward, right, right-backward, backward, left-backward, and left. The Turn left recognizer was implemented symmetrically. As the users performed this gesture with quite different movement speeds, the recognizer was quite tolerant to different speeds. However, it required the circle to be drawn smoothly to cover all of the six required directions with a recognizable long enough period of time.

The Stop movement action was implemented with a recognizer for one hand to be stretched to the front with open fingers. Therefore, it looked at the relation between the shoulder and the hand joint to ensure the correct hand position and, in addition, it applied a finger count recognizer for recognizing an open hand. The finger count recognizers in FUBI works directly on the depth data around the hand joint and attempts to separate displayed fingers from the rest of the hand by applying a morphological opening operation similar to the process described by [45]. A visualisation of the extracted finger shapes during the gesture for the Stop movement action can be seen in Fig. 4d. The recognizer for the gesture required a minimum of three recognized displayed fingers as can be seen in the XML definition depicted by Fig. 3. This was found to be robust enough to ignore the fingers pointing to the front or forming a fist and to detect an open hand in which some fingers are relatively close to each other. For the timing, a minimum of one second was required for the hand to stay in front, and, at the end of that phase, the detection for an open hand needed to be fulfilled for at least half a second. The last period is a bit shorter than the timings we extracted from the video performances. It was chosen this way, as the finger count recognizer often suffers

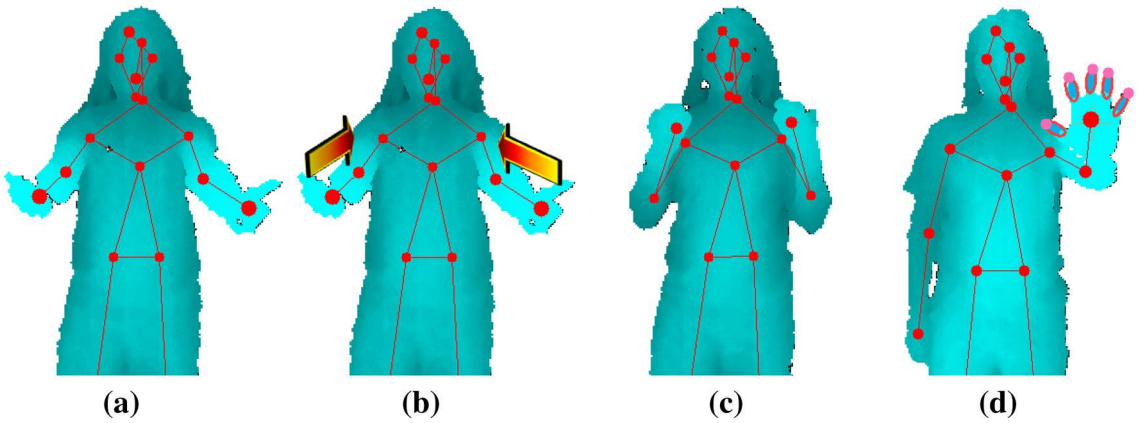


Fig. 4 Tracking image for the different recognition steps of the actions *Speed up* (**a**–**c**) and *Stop movement* (**d**)

from noise in the depth or tracking data and we got a more reliable recognition in this way.

The recognizers for the actions Speed up, Slow down, and Stand up were implemented in a similar way. They all required one or both hand/s in a specific starting position. Then waited for a movement in backward (Speed up), downward (Slow down), or upward (Stand up) directions. After that they expected a specific end position of the hand/s. For example, the action Speed up requires a starting position of both hands at least one shoulder width in front of the body (Fig. 4a). After a short movement in backward direction (Fig. 4b), the hands should be closer than one hip width to the body (Fig. 4c). The timings were chosen in a way that the whole gestures took at least 0.5–0.7 s as a lot of study participants performed the movement quite fast.

5.3 Results

We conducted an evaluation to test for the accuracy performances of the proposed recognition system. Eleven gestural commands were implemented in the recognition system as described in Sect. 5.2. We recruited 22 participants (7 female and 15 male) with an average age of 26 (SD = 4.7) and they are all right handed. As we are testing for the accuracy of classifying predefined gestures in the system, the participants first practiced each gesture, thus, their background did not matter when selected for participation.

The experiment was arranged in a room about 3 meters wide and 6.5 meters in depth. It is equipped with a 50 inch plasma display and a Kinect device was placed in the centre, directly below the display. The following were the steps each participant is asked to follow:

1. When the participants enter the room, they are instructed by the administrator to stand at a point about 2 meters away from the Kinect device and a description of the study is given to them. All tests are performed from the 2 meters mark, except for the gesture of the Stop movement action, where the user is asked to stand about 1.5 meters away from the screen.
2. The participants are asked to watch a video demonstrating one of the body gestures of the eleven actions. The participant practiced the gesture and they were allowed to watch the video as many times as they want.
3. When the participants say that they are ready and understand the gesture they watched, the administrator instructs them to perform that gesture.
4. After the first performance, the participants are asked to return to their starting point (if necessary), and repeat the gesture one more time.
5. All participants continues with step 2 until the gestures for all actions have been performed twice.
6. When completing the experiment, the participant is asked to fill out a short questionnaire

The eleven actions are presented to each participant in a weak counter-balanced order (Latin Square order) and data is collected by the system automatically using the recognition system, described in Sect. 5.2. The recognition system therefor run in real-time on a standard notebook computer, however, there was no robot present that would have reacted on the gestures directly, but the recognition results were only logged for the evaluation. This was done to not distract the users from their gesture performance and to avoid having them repeat a failed gesture until the robots reacts.

In total, 484 body gestures/ body motions were analyzed by our system and the overall results are summarized in Table 2. For each action, it lists the implemented gesture candidate and the recognition results. Those include the number of correctly classified gestures = true positives (TP), the number of falsely classified gestures = false positives (FP), and the gestures our system did not recognize at all = false negatives (FN). From these three values, we further calculated the

precision $\left(= \frac{TP}{TP+FP}\right)$, accuracy $\left(= \frac{TP}{TP+FP+FN}\right)$, and recall $\left(= \frac{TP}{TP+FN}\right)$.

The results show that the overall recognition accuracy of the system for all actions was 74 %, with the highest recognition rate of the Slow down action (96 %) and the lowest rate of the Speed up action (12 %). A similar result can be seen for the recall of our system. The problem with the Speed up action was that we accidentally used the recognizer implemented for recognition with tracking data by NiTE instead of the Kinect SDK tracking. The gesture for Slow Down was defined by both arms stretched to the front and then coming close to the body as visualized in Fig. 4, first three images from the left. For recognizing that the hands are close, we compared the distance along the z-axis between the hands and the torso joint (called spine in the Kinect SDK). The maximum for that distance was defined as the hip width, which is unfortunately much smaller for the Kinect SDK (see Fig. 4) compared to NiTE. In addition, the torso joint in the Kinect SDK is positioned a bit more backwards. Finally, when the hands are close to the body, the Kinect SDK usually reports a reduced tracking confidence, but we requested a higher confidence in our recognizer as it would be reported by the NiTE tracking. Therefore, the recognizer failed in most times to detect that the hands are close to the body.

The recognition of the Turn left/right actions showed a much lower accuracy than the other actions. It seems that the recognizers were defined in a strict way, which caused the high number of false negatives.

The average accuracy of the system improves to 80 % by omitting the Speed up action, and to 89 % by additionally omitting the actions Turn right and left.

Regarding the remaining actions, Stop movement was recognized with a slightly lower accuracy (73 %) than the others. This was due to the fact that it was the only action including finger count recognizers that are in general less accurate then the other recognizers as they suffer more heavily from the distortions in the depth stream. This caused a slightly higher number of false negatives for this action.

The precision of our system was 89 % on average (93 % without Speed up) and herein, also the gestures for the actions Turn right and left get an acceptable number. Therefore, if our system detected a gesture performance, it usually detected the correct one. Only the Speed up action provides a low precision for the same reasons a mentioned above, all other actions have at least a precision of 83 %, with some of them even reaching 100 %.

6 Conclusion and Future Work

In this paper, we have presented the results of a study to produce a user-defined gesture set to navigate a humanoid robot intuitively. The study yield to the development of a full body recognition system that can be used to classify the defined gestures.

To define the users' preferences in navigating a humanoid robot using gestural commands, we conducted a study on 35 participants that belong to two groups: technology aware users (i.e. gesture recognition and robots), and non-experienced users. The analysis of the data revealed (1) a user-defined gesture set to control a humanoid robot, (2) a taxonomy of the human-robot navigational gestures, (3) user agreement scores for each of the gestures representing a navigational commands, (4) time performances of the gesture motions, and (5) design implications for gesture recognition.

Based on the results of the study, we developed a recognition system for classifying the user-defined gestures using our open source Full Body Interaction Framework (FUBI).[8] The presented recognition system was evaluated by 22 participants to achieve an average classification rate of 74 %. However, the accuracy of the system improved up to 89 % when omitting gestures with systematic errors. This achieved accuracy clearly shows encouraging results and can lead to effectively using the system in navigating a humanoid robot.

To have a complete framework for the gestural control of humanoid robots, we will need to improve some of the recognizers and integrate the rest of our user-defined gesture set, so that users are able to configure which gestures they want to use for a specific command. As the FUBI framework provides an easy way to define own gestures, users could further completely customize the gesture set to their preferences. By editing the XML gesture definitions, present gestures can be modified and new ones can be added as well. Another option to personalize the gesture set would be to let users record their own gestures to train the system directly, or to adapt and refine the gestures while the users already interact with the robot to improve the recognition. We will further investigate other recognition techniques to compare them with FUBI.

The presented work covers gestures presented by participants with a German cultural background. However, Kita et al. [46] gives a review of cross-cultural variations of gestures and outlines factors for those variations. Moreover, Bartneck et al. [47,48] and Nomura et al. [49], discuss how culture does have an impact on the how we perceived robots. This indeed opens up future research work in the area of designing universal gestures to control humanoid robots. This requires several collaborations from different countries in several continents. Thus, our work can serve to be the bases of such future research.

In the presented study, we focused on navigational commands, however, a humanoid robot can do more functions that can be also investigated in future work. In addition, the

[8] http://www.hcm-lab.de/fubi.html.

subjective study revealed that a combination between gesture and speech commands is important and will be investigated in future work.

Acknowledgments This work was partially funded by the European Commission within the 7th Framework Program under grant agreement eCute (FP7-ICT-257666).

References

1. Kistler F, Endrass B, Damian I, Dang C, André E (2012) Natural interaction with culturally adaptive virtual characters. J Multimodal User Interfaces 6:39–47
2. Suma EA, Lange B, Rizzo A, Krum DM, Mark B (2011) FAAST: the flexible action and articulated skeleton toolkit. In: Proceedings of the virtual reality, Singapore, pp 47–248
3. Stiefelhagen R, Fugen C, Gieselmann R, Holzapfel H, Nickel K, Waibel A (2004) Natural human-robot interaction using speech, head pose and gestures. In: Proceedings of cthe IEEE/RSJ international conference on intelligent robots and systems, (IROS 2004), 3:2422–2427
4. Suay HB, Chernova S (2011) Humanoid robot control using depth camera. In: Proceedings of the 6th international conference on Human-robot interaction, HRI '11. NY, USA, ACM, New York, pp 401–402
5. Wobbrock JO, Morris MR, Wilson AD (2009) User-defined gestures for surface computing. In: Proceedings of the 27th international conference on Human factors in computing systems, CHI '09. NY, USA, ACM, New York, pp 1083–1092
6. Kurdyukova E, Redlin M, André E (2012) Studying user-defined ipad gestures for interaction in multi-display environment. In: International Conference on Intelligent User Interfaces, ACM, New York, pp 1–6
7. Häring M, Eichberg J, André E (2012) Studies on grounding with gaze and pointing gestures in human-robot-interaction. In: Ge ShuzhiSam, Khatib Oussama, Cabibihan John-John, Simmons Reid, Williams Mary-Anne (eds) Social robotics, vol 7621 Lecture notes in computer science. Berlin Heidelberg, Springer, pp 378–387
8. Maha S, Rohlfing K, Kopp S, Joublin F (2011) A friendly gesture: Investigating the effect of multimodal robot behavior in human-robot interaction. In: IEEE, RO-MAN, Atlanta, 3: 247–252
9. Sidner CL, Lee C, Kidd CD, Lesh N, Rich C (2005) Explorations in engagement for humans and robots. Artif Intell 166(12):140–164
10. Salem M, Eyssel F, Rohlfing K, Kopp S, Joublin F (2013) To err is human(-like): effects of robot gesture on perceived anthropomorphism and likability. Int J Soc Robot 5(3):313–323
11. Salem M, Kopp S, Wachsmuth I, Rohlfing K, Joublin F (2012) Generation and evaluation of communicative robot gesture. Int J Soc Robot 4(2):201–217
12. Efron D (1941) Gesture and Environment. King's Crown Press, Morningside Heights, New York
13. Ekman P, Friesen W (1969) The repertoire of nonverbal behavior: categories, origins, usage, and coding. Semiotica 1:49–98
14. McNeill D (1985) So you think gestures are nonverbal? Psychol Rev 92(3):350–371
15. McNeill D (1992) Head and mind: what gestures reveal about thought. University of Chicago University of Chicago Press, Chicago
16. McNeill D (2005) Gesture and thought. University of Chicago Press, Chicago
17. Saffer D (2009) Designing gestural interfaces. O'Reilly Media, Sebastopol
18. Jaime R, Yang L, Edward L (2011) User-defined motion gestures for mobile interaction. In: Proceedings of the 2011 annual conference on Human factors in computing systems, CHI '11. NY, USA, ACM, New York, pp 197–206
19. Christian K, Daniel N, John D, Michael R (2010) User-defined gestures for connecting mobile phones, public displays, and tabletops. In: Proceedings of the 12th international conference on Human computer interaction with mobile devices and services, MobileHCI '10. NY, USA, ACM, New York, pp 239–248
20. Zhang L, Huang Q, Liu Q, Liu T, Li D, Lu Y (2005) A teleoperation system for a humanoid robot with multiple information feedback and operational modes. In: IEEE international conference on robotics and biomimetics (ROBIO), pp 290–294
21. Kechavarzi BD, Sabanovic S, Weisman K (2012) Evaluation of control factors affecting the operator's immersion and performance in robotic teleoperation. In: IEEE, RO-MAN, pp 608–613
22. Sian NE, Yokoi K, Kajita S, Kanehiro F, Tanie K (2002) Whole body teleoperation of a humanoid robot - development of a simple master device using joysticks. In: IEEE/RSJ international conference on intelligent robots and systems, vol. 3, pp 2569–2574
23. McColl D, Zhang Z, Nejat G (2011) Human body pose interpretation and classification for social human-robot interaction. Int J Soc Robot 3(3):313–332
24. Sakagami Y, Watanabe R, Aoyama C, Matsunaga S, Higaki N, Fujimura K (2002) The intelligent ASIMO: system overview and integration. In: IEEE/RSJ international conference on intelligent robots and systems, vol. 3, pp 2478–2483
25. Yorita A, Kubota N (2011) Cognitive development in partner robots for information support to elderly people. IEEE Trans Auton Ment Dev 3(1):64–73
26. Ju Z, Liu H (2010) Recognizing hand grasp and manipulation through empirical copula. Int J Soc Robot 2(3):321–328
27. Fujimoto I, Matsumoto T, Silva PRS, Kobayashi M, Higashi M (2011) Mimicking and evaluating human motion to improve the imitation skill of children with autism through a robot. Int J Soc Robot 3(4):349–357
28. Yun S-S, Kim M, Choi MT (2013) Easy interface and control of tele-education robots. Int J Soc Robot 5(3):335–343
29. Waldherr S, Romero R, Thrun S (2000) A gesture based interface for human-robot interaction. Auton Robot 9(2):151–173
30. Nguyen-Duc-Thanh N, Stoniern D, Lee SY, Kim DH (2011) A new approach for human-robot interaction using human body language. In: Proceedings of the 5th international conference on convergence and hybrid information technology, ICHIT'11. Springer, Berlin, pp 762–769
31. Broccia G, Livesu M, Scateni R (2011) Gestural interaction for robot motion control. In: EuroGraphics Italian chapter, pp 61–66
32. Cabibihan J-J, So W-C, Pramanik S (2012) Human-recognizable robotic gestures. IEEE Trans Auton Mental Dev, 4(4):305–314
33. Strobel M, Illmann J, Kluge B, Marrone F (2002) Using spatial context knowledge in gesture recognition for commanding a domestic service robot. In: Proceedings of the 11th IEEE international workshop on robot and human interactive communication, pp 468–473
34. Sato E, Yamaguchi T, Harashima F (2007) Natural interface using pointing behavior for human-robot gestural interaction. In: IEEE transactions on industrial electronics, 54(2):1105–1112
35. Sato E, Nakajima A, Yamaguchi T, Harashima F (2005) Humatronics (1)— natural interaction between human and networked robot using human motion recognition. In: IEEE/RSJ international conference on intelligent robots and systems, (IROS 2005), pp 930–935
36. Hu C, Meng MQ, Liu PX, Wang X (2003) Visual gesture recognition for human-machine interface of robot teleoperation. In: IEEE/RSJ international conference on intelligent robots and systems, (IROS 2003). Proceedings, vol. 2, pp 1560–1565
37. Konda KR, Königs A, Schulz H, Schulz D (2012) Real time interaction with mobile robots using hand gestures. In: Proceedings of

the seventh annual ACM/IEEE international conference on human-robot interaction, HRI '12. NY, USA, ACM, New York, pp 177–178
38. Dillmann R (2004) Teaching and learning of robot tasks via observation of human performance. Robot Auton Syst 47(23):109–116
39. Breazeal C, Scassellati B (2002) Robots that imitate humans. Trends Cogn Sci 6(11):481–487
40. Barattini P, Morand C, Robertson NM (2012) A proposed gesture set for the control of industrial collaborative robots. In IEEE RO-MAN, pp 132–137
41. Ende T, Haddadin S, Parusel S, Wusthoff T, Hassenzahl M, Albu-Schaffer A (2011) A human-centered approach to robot gesture based communication within collaborative working processes. In: IEEE/RSJ international conference on intelligent robots and systems (IROS), pp 3367–3374
42. Gleeson B, MacLean K, Haddadi A, Croft E, Alcazar J (2013) Gestures for industry intuitive human-robot communication from human observation. In: 8th ACM/IEEE international conference on human-robot interaction (HRI), pp 349–356
43. Bodiroa S, Stern HI, Edan Y (2012) Dynamic gesture vocabulary design for intuitive human-robot dialog. In: 7th ACM/IEEE international conference on Human-robot interaction (HRI), pp 111–112
44. Wobbrock JO, Aung HH, Rothrock B, Myers BA (2005) Maximizing the guessability of symbolic input. In: CHI '05 extended abstracts on Human factors in computing systems, CHI EA '05, ACM, New York, pp 1869–1872
45. Kang SK, Nam MY, Rhee PK (2008) Color based hand and finger detection technology for user interaction. In: ICHIT '08. international conference on convergence and hybrid information technology, pp 229–236
46. Kita S (2009) Cross-cultural variation of speech-accompanying gesture: a review. Lang Cogn Process 24(2):145–167
47. Bartneck C, Nomura T, Kanda T, Suzuki T, Kato K (2005) Cultural differences in attitudes towards robots. In: Proceedings of the symposium on robot companions: hard problems and open challenges in Robot-human interaction,
48. Bartneck C, Suzuki T, Kanda T, Nomura T (2007) The influence of people's culture and prior experiences with aibo on their attitude towards robots. AI Soc 21:217–230
49. Nomura T, Suzuki T, Kanda T, Han J, Shin N, Burke J, Kato K (2008) What people assume about humanoid and animal-type robots: cross-cultural analysis between japan, korea, and the united states. Int J Hum Robot 05(01):25–46

Mohammad Obaid is a postdoctoral fellow at the t2i Lab, Chalmers University of Technology. In 2007, he gained his MSc. degrees from the Computer Science and Software Engineering Department, at the University of Canterbury, New Zealand, in which he gained First Class Honours. In 2011, he gained his PhD. degree in Computer Science and Software Engineering from the University of Canterbury, New Zealand. He started his career (2011–2014) with a postdoctoral fellowship at both the Human Centered Multimedia Lab (Augsburg, Germany) and the Human Interface Technology Lab New Zealand (HITLabNZ). He collaborated and worked at several international research institutes including CNRS at LTCI, Telecom ParisTech (Paris, France), Australian National University (Canberra, Australia), Institute for Computer Graphics and Vision at the Graz University of Technology (Graz, Austria), and the Digital Media department at the Upper Austria University of Applied Sciences (Hagenberg, Austria). His research interests fall in the areas of Human-Computer Interaction and Human-Robot Interaction.

Felix Kistler graduated in 2010 at the Augsburg University and joined the Human Centered Multimedia directly afterwards. As a student, he helped in the development of the labs GameEngine that he is currently still maintaining. After an internship at the game development studio Related Designs (Mainz, Germany), where he worked on the popular strategy game Anno 1404, he implemented and evaluated a level of detail based behavior control for intelligent virtual characters in his diploma thesis. As a PhD student, he started working on novel interaction techniques, especially in combination with depth sensors. His work led to the development of the Full Body Interaction framework (FUBI) (http://www.hcm-lab.de/fubi.html) which was mainly developed for usage in the EU project eCute.

Markus Häring graduated as a Master of Science in Informatics and Multimedia from Augsburg University, Germany in 2010. As a student researcher he worked on dialog modelling for virtual characters in the EU project DynaLearn. Afterwards, he started his PhD at the lab for Human Centered Multimedia with a focus on collaborative human-robot interaction. His research interests also include usability engineering, multimodal interaction, and social robotics.

René Bühling graduaded at Augsburg University of Applied Sciences and holds two diplomas and two master degrees in Computer Science and Multimedia Design, focussing on Game Development. Since 2009 Ren is employed as research assistant at the Lab for Human Centered Multimedia at the University of Augsburg, Germany. His PhD research interests include Aesthetic Technology, Visual Narratives, Character Design, Entertainment Computing, and synergies of Visual Arts and Computer Technology in general. He is teaching 3D Design and Character Design and was working on several national and international research projects.

Elisabeth André is a full professor of computer science at Augsburg University and chair of the Laboratory for Human Centered Multimedia, Augsburg University, Augsburg, Germany. Earlier, she worked as a principal researcher at DFKI GmbH, where she led various academic and industrial projects in the area of intelligent user interfaces. In summer 2007, she was nominated a Fellow of the Alcatel-Lucent Foundation for Communications Research. In 2010, she was elected a member of the prestigious German Academy of Sciences Leopoldina and the Academy of Europe. Her research interests include affective computing, intelligent multimodal interfaces, and embodied agents.