**Noname manuscript No.**
(will be inserted by the editor)

# Gestures for Manually Controlling a Helping Hand Robot

**Mahisorn Wongphati · Hirotaka Osawa · Michita Imai**

Received: date / Accepted: date

**Abstract** Helping hand robots have been the focus of a number of studies and have high potential in modern manufacturing processes and for use in daily living. As helping hand robots interact closely with users, it is important to find natural and intuitive user interfaces for interacting with the robots in various situations. This study describes a set of gestures for interacting with and controlling helping hand robots in situations in which users need to manually control the robot but both hands are not available, for example, when users are holding tools or objects in their hands. The gestures are derived from an experimental study that asked participants for gestures suitable for controlling primitive robot motions. The selected gestures can be used to control translation and orientation of an end effector of a helping hand robot when one or both hands are engaged with tasks. As an example for validating the proposed gestures, we implemented a helping hand robot system to perform a soldering task.

**Keywords** gesture · helping hand · human–robot interaction · user-defined · human–robot collaboration

M. Wongphati
Graduate School of Science and Technology
Keio University, Japan
E-mail: mahisorn@ayu.ics.keio.ac.jp

H. Osawa
Faculty of Intelligent Interaction Technologies
Tsukuba University, Japan
E-mail: osawa@iit.tsukuba.ac.jp

M. Imai
Faculty of Science and Technology
Keio University, Japan
E-mail: michita@ayu.ics.keio.ac.jp

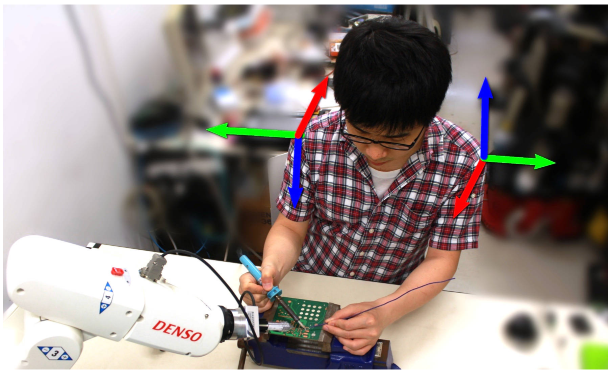


**Fig. 1** The user is trying to solder the cable in his left hand to a circuit board while holding the soldering iron in his right hand. The robot moved to a position near a soldering point. However, the user found that it was not in the position he expected and had to manually control the end effector with body movement gestures to correct the error.

## 1 Introduction

This study proposes a set of gestures for manually controlling an end effector when working closely with an industrial robot that acts as a helping hand. We focused mainly on a situation when the user's hands cannot be used or moved because of the task's demands. For example, when the user is holding tools and objects in place and waiting for help from the robot, a situation such as that depicted in Fig. 1 ensues. The proposed gestures allow the user to precisely adjust the position and orientation of an end effector without interruption (e.g., releasing the soldering iron or the cable) using body movement and hand gestures when the robot cannot perform the expected operation.

Here is an example scenario. A user is trying to solder a cable held in the left hand to a circuit board while

(1) U: “Solder wire”
(2) R: “Okay, please wait”
(3) R: “Where do you want me to add the solder wire?”
(4) U: “Here”
(5) R: “Okay”
(6) U: “A little bit to the left”
(7) U: “A little bit to the right”
(8) U: (repeats (6) and (7) several times)
(9) U: “Follow me”

**Fig. 2** The dialog between the user and the robot in the example scenario.

holding a soldering iron in the right hand. How does the user interact with a helping hand robot to tell it to feed the solder wire to a specific soldering point? The dialog presented in Fig. 2 reflects one of the possible scenarios (U: user, R: robot).

In line (1), the user requests a solder wire from the robot and the robot responds with line (2). After the solder wire has been prepared, the robot asks for a soldering point in line (3) and the user responds as in line (4). Because the user is working on a prototype circuit board for testing a new design and the robot does not know about the layout and exact positions of components and soldering pads, the robot has to perform its best guess based on limited sensor information by, for example, detecting the location of the user’s hands and tip of the soldering iron for interpreting the “here” position in line (4). The robot then responds with an utterance in line (5) and moves the tip of the solder wire to the guessed position. However, the guessed position is different from the one that user is expecting and the user has to correct the error by repeatedly issuing the utterances (6) and (7) until the user gives up and asks the robot to move according to user’s gestures with the command in line (9) to manually control the helping hand robot.

The example scenario shows the need of an additional method for the user to manually control the end effector of a helping hand robot in certain situations. Traditional methods, such as teaching pendants or joysticks are possible choices, but they are usually cumbersome, in particular when both hands are engaged with tasks. This constraint also makes tactile control, such as a force–torque sensor or a joint impedance control that allows users to manipulate the end effector directly become an unfeasible choice. Although controlling with verbal commands is one of the possible choices for commanding the robot, the verbal commands are usually error-prone when they are used to convey spatial information that contains deictic terms such as “here”, “that one”, and “over there” [16]. Both input devices and verbal commands are also tediously repetitive when used as depicted in line (8) in the scenario dialog.

On the other hand, controlling with gestures is more suitable for this type of situation because gestures are a less ambiguous method for conveying spatial information to a computer system [22] and have already been adopted in a number of studies [12] [29].

This study derived a set of gestures from a previous experimental study [34] and implemented a helping hand robot system that can be manually controlled with the gestures. We implemented a gesture recognition module for testing and refining the derived gestures. The robot system was customized as a helping hand tool for a soldering task that requires handiness and dexterity of both the human and the robot. Results from this study are intended for complementing the mentioned methods to emphasis the need of suitable multi-modal communication channels for the helping hand robot [30].

The rest of the paper is organized as follows. Section 2 depicts an overview of related work. Section 3 presents the robotic helping hand system and its components. Section 4 describes the development of gestures and recognition methods. Section 5 explains the experiment to show the usefulness and intuitiveness of the proposed gestures. The discussion and conclusion are summarized in sections 6 and 7, respectively.

## 2 Related Work

After the introduction of industrial robots in the 1960s [20], Many studies have shown the potential uses of a robot arm as a helping hand in the healthcare domain since the early days [15]. When robots became cheaper because of demand and because the cost of skilled labor increased, the concept of combining the precision and repeatability of a robot with the problem-solving skills of humans to create flexible manufacturing processes and a flexible working environment was introduced. The main focus of the concept is to make it easier for humans to program and work closely with robots [6] [14].

One of the noticeable efforts in human–robot interaction (HRI) and human–robot collaboration (HRC) development is the integration of multimodal communication and user interfaces into robotic systems to allow both experts and novices to communicate more easily, to program and configure robots according to their preferences and task requirements [6].

From various communication and interaction methods, gestures have been selected by a number of studies as a flexible and natural method for communication between human and robot systems in various situations [12].

Rogalla et al. [24] presented a method for using hand gestures for commanding a mobile manipulator robot.

A set of gestural commands was defined for the robots, such as yes/no, stopping, and grasping objects.

Iossifidis et al. [17] proposed a design concept of anthropomorphism in an assistant robot. Users can interact with the robot via multimodal communication including gestures for demonstrating a simple assembly task.

Stiefelhagen et al. [26] proposed multimodal interaction for HRI with speech, head pose, and gestures. The study used pointing gesture and head pose (face direction) to recognize users' desired targets or objects.

Wachs et al. [28] proposed a reconfigurable gesture recognition system and demonstrated it by teleoperating a robot manipulator with a set of predefined gestures.

Tran et al. [27] proposed a wireless data glove that has various motion sensors such as an accelerometer, gyros, and magnetic sensors to detect hand gestures for controlling various functions of a military robot.

Wallhoff et al. [30] presented a hybrid assembly station that enabled human workers to teach and interact with a helping hand using speech, gaze, and tactile input from the projected user interface.

Burger et al. [8] summarized an effort to develop a mobile manipulator platform that can be commanded with speech commands and two-handed gestures and provides a comprehensive review on related work about hand gestures in HRI research.

With advancements in computing power and sensors such as RGB-D sensors (e.g. Kinect [3]), complex hand gestures can be recognized in real time (e.g. Wang et al. [31] and Oikonomidis et al. [21]).

Surprisingly, an option for manually controlling helping hand robots while working closely with them was not formally discussed in the mentioned studies, despite the fact that it is the last resort for users to overcome a glitch with their problem-solving skills when the robot's performance does not meet their expectation, as shown in the example scenario (Fig. 2).

Gestures in the mentioned works were usually designed by system developers who were familiar with the system's capability. The designers tended to select gestures based on ease of detection and distinguishability to increase recognition reliability. Therefore, the developer-designed gestures might not represent the real expectations of users and might feel unnatural to the users [33].

This intuition has led to the studies about user-defined gestures for robot systems from Wongphati et al. [34] and Gleeson et al. [11]. The studies focus on finding user-defined gestures for manually controlling and communicating with robots during an HRC session through user-centered design methodology.

In [34], the authors conducted an experimental study to collect user-defined gestures for controlling the basic movements (up, down, left, right, forward, and backward) of an end effector of a virtual robot in a simulated soldering task. From the study, when gestures were articulated while both the user's hands were occupied, one with a soldering iron and the other with a cable, the following findings delivered important constraints for gesture selection and implementation of a gesture recognition system.

- Hand (one or both hands) and body movement (e.g. tilt, lean, or twist body) gestures are the dominant gestures.
- Participants who were holding objects in their hands from the start of the task would perform gestures without releasing the objects they were holding.
- Most participants used the left or right or both hands interchangeably for articulating hand gestures.
- Reversible gestures for controlling basic movement such as left and right by sweeping the hand to the left and right were consistently performed by the participants.
- Body movement gestures were articulated only by the participants who were holding their hands in a working pose as shown in Fig. 1.

Trying to manually control an end effector with six degrees-of-freedom (DOFs) is not a trivial process and therefore we observed methods used by various systems to select and prepare a set of suitable gestures. A teaching pendant for a robot arm handles this issue by providing separated control for each DOF (Fig. 3(a)). Three-dimensional (3D) computer aided design (CAD) and computer aided machining (CAM) software provide an option for decoupling the translation from orientation control when manipulating objects within the software (Fig. 3(b)). Humans also usually transfer or move objects to a desired destination before/after aligning their orientation [18].

These observations and the trial-and-error testing with various control methods such as a 3D interactive marker in ROS [23], a 6-DOF 3D mouse [5], and a tactile control that utilizes a force–torque sensor on the implemented helping hand robot give the following intuitive ideas about how to manually control an end effector.

- It is easier to control translation and orientation separately.
- Translation and orientation control should be able to switch between workspace and end effector (tool) frames.
- Controlling the orientation of an end effector in the workspace frame is not intuitive.

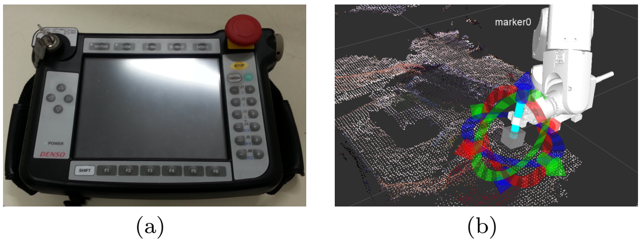


**Fig. 3** (a) The teaching pendant for an industrial robot from Denso Corporation. Each joint of the robot can be controlled using two rows of buttons, at left. (b) A 6-DOF 3D interactive marker in the robot operating system (ROS) that can be manipulated with a mouse by pulling arrows for translation or dialing rings for orientation control.

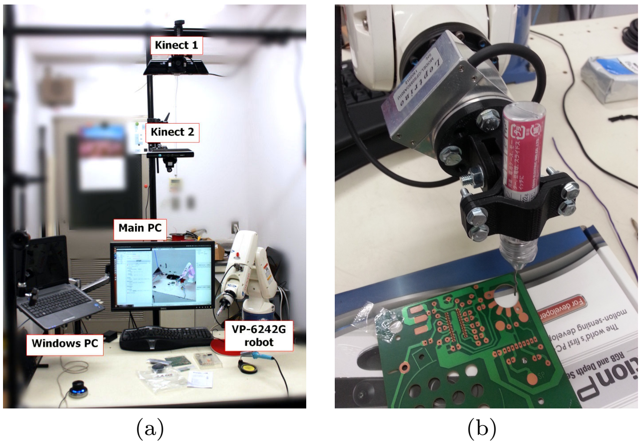


**Fig. 4** (a) The helping hand robot system for the soldering task. (b) The 3D printed solder wire holder.

- An individual axis *(x, y, z)* or plane *(x–y, x–z, y–z)* should be selectable for translation control.
- Orientation control is more intuitive and easier to handle when each axis (roll, pitch, yaw) is controlled separately.

## 3 The Helping Hand Robot System

### 3.1 Hardware and Software

The helping hand robot is a 6-DOF Denso VP-6424G industrial robot mounted on a table (Fig. 4).

Two Kinect cameras are used as main sensors. The first camera (Kinect 1 in Fig. 4(a)) is mounted over the workspace and connected to the main PC. Its raw point cloud data is used for workspace calibration, object and arm detection, and hand gesture recognition. A point cloud library (PCL) [25] is used for processing the point cloud data. bThe second camera is introduced because the upper body of the user cannot be seen in the first camera field of view.

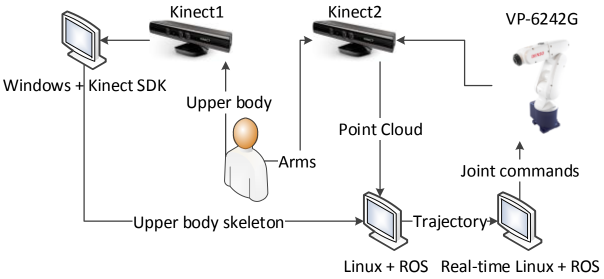


**Fig. 5** Overview of the components of the helping hand robot system.

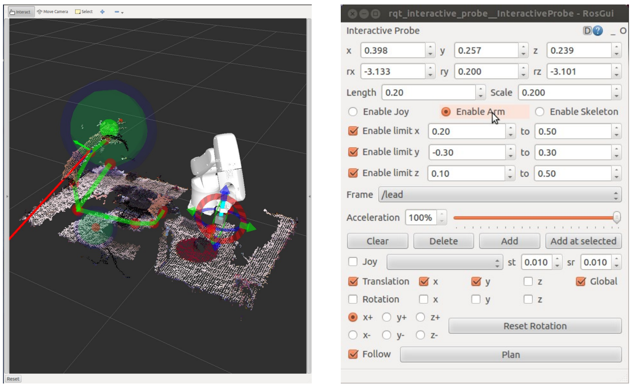
The second camera (Kinect 2 in Fig. 4(a)) is mounted in front of the workspace and pitched downward for detecting the upper body. It is used for recognizing upper body skeletons with Microsoft Kinect SDK [3] that available only on a Windows PC. The recognized skeleton information (e.g., joint positions) is sent to a main PC for gesture recognition calculations to be performed. Detailed information about gesture recognition and algorithm will be discussed in Section 4.2.

The main PC is a Linux system with a ROS that handles all interactions between the user and the helping hand robot. After the target position of the end effector of the robot is computed from the interaction between the user and the system, trajectories of the robot are generated and sent to a real-time Linux PC to be converted to joint commands and transmitted to the robot controller at 1000 Hz. The need of the separated real-time Linux PC is caused by the computation load of the main PC. The load prevents the main PC from sending trajectory commands with less than 2 ms jitter which is required by the robot controller.

A diagram of the overall system is shown in Fig. 5. All source codes of the implemented system are open-source and available at [2].

### 3.2 User Interface

The main screen of the user interface (UI) is based on the 3D visualization tool for ROS (rviz) [1] shown in Fig. 6(a). In rviz, users can perform all common 3D interface controls such as pan, tilt, zoom, and rotate the scene to match their preferences and controlling methods. Robot states such as the positions of joints are updated in real time with data from the robot controller and displayed with a 3D model of the robot in rviz (a white mesh in Fig. 6(a)). The real-time updated robot model is also used as a supplementary virtual feedback for gestures and state of the robot for the participants during the experiment.



rqt_interactive_probe__InteractiveProbe - RosGui

Interactive Probe

x 0.398 y 0.257 z 0.239

rx -3.133 ry 0.200 rz -3.101

Length 0.20 Scale 0.200

○ Enable Joy ● Enable Arm ○ Enable Skeleton

☑ Enable limit x 0.20 to 0.50

☑ Enable limit y -0.30 to 0.30

☑ Enable limit z 0.10 to 0.50

Frame /lead

Acceleration 100%

Clear Delete Add Add at selected

□ Joy st 0.010 sr 0.010

☑ Translation ☑ x ☑ y □ z ☑ Global

□ Rotation □ x □ y □ z

● x+ ○ y+ ○ z+ Reset Rotation

○ x- ○ y- ○ z-

☑ Follow Plan

(a) The rviz screen (b) The control panel

**Fig. 6** Screen shots of the implemented user interface, which is displayed on a large monitor behind a workspace (Fig. 4(a)) to allow a user to control and monitor all system states.

An interactive marker in rviz (magnified in Fig. 3(b)) is used to manually control or set up an end effector of the robot [13]. The user can drag an arrow or dial a ring to perform translation or orientation control, respectively. The interactive marker is used as one of the testing conditions in the experiment in Section 5. Obstacle avoidance, self-collision checking, and invert kinematic solving are based on the MoveIt! library [4].

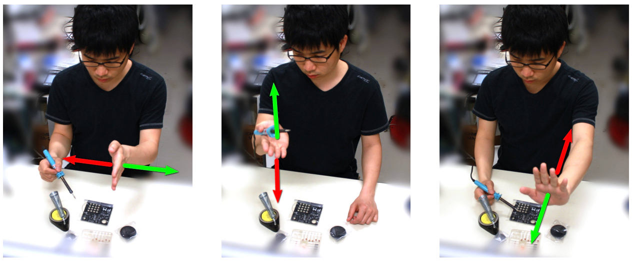
A control panel in Fig. 6(b) is mainly used for setting up the robot system and selecting the interaction mode (3D markers, hand gestures, or body movement gestures) during the development and during the experiment by an instructor in Section 5. Users can also select a desired working frame (end effector or workspace frames), axis *(x,y, or z)*, or plane *(x–y, x–z, or, y–z)* on the control panel while interacting with the helping hand robot.

## 4 Gesture Development

### 4.1 Gestures

In this study, we focused mainly on how to allow users to manually control an end effector when one or both hands are engaged with other tasks. With suggestions from [34], we selected body movements and hand gestures for manually controlling the translation and orientation of the end effector.

To be more specific with the manual control, we divide the manual control into two steps. The first step is setting up an end effector (e.g. moving the end effector from its initial position to a target working area). The second step is controlling the end effector around the working area based on the task's requirements.



(a) Left/Right (b) Up/Down (c) Front/Back

**Fig. 7** One-handed gestures for translation control. The user can use either the left or right hand to control an end effector while holding objects in the other hand.

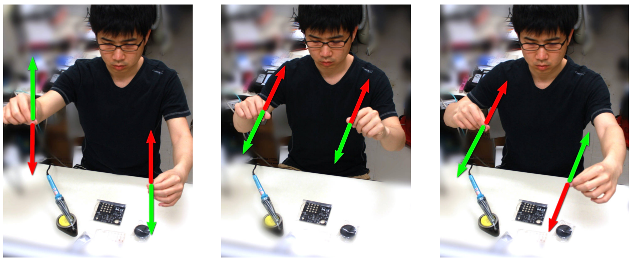
The first step can be seen as a rough control step that requires speed over precision, for example, moving the end effector from the rightmost side of the workspace to a circuit board (working area) in the middle of the workspace (Fig. 13(a)). On the other hand, the second step requires precision over speed when both hands are engaged with tasks such as moving the tip of the solder wire into a soldering point (Fig. 1).

Body movement gestures as shown in Fig. 1 allow the users to handle the second step by controlling translation motions of an end effector without interruption while dealing with tasks with both hands. Although body movements are suitable for precision control, they are limited by working postures (e.g. being seated) and are not convenient for the first step, which usually deals with displacement over a large distance.

To overcome limitations of the body movement gestures, one- and two-handed gestures as shown in Fig. 7 and 8 have been selected for the first step to allow the users to control the translation and orientation of an end effector with the left and/or right hand from any position in the workspace. For the orientation control, a combination of left- and right-hand gestures that resemble the action of holding a sheet of paper in both hands and flipping or rotating the paper around the $x$, $y$, or $z$ axes were chosen based on gestures for CAD systems from [31].

It is possible for the user to control both translation and orientation with only body movement gestures (e.g. twisting the body for controlling the yaw motion of the end effector). However, from our preliminary testing, we found that it is inconvenient and difficult to maintain good eye–hand coordination when compared with the body movement gestures that employ a simple tilting of the body for translation control. The eye–hand coordination is important for safety and task quality when working closely with the robot.

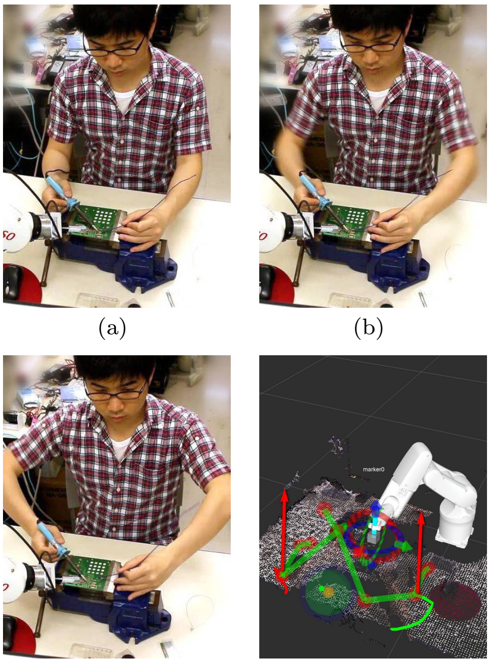
Furthermore, there is a requirement for toggling between the workspace and the end effector frames while



(a) Roll (b) Pitch (c) Yaw

**Fig. 8** Two-handed gestures for orientation control. The users can use both hands to control the orientation of an end effector with a "sheet of paper" metaphor while holding objects in both hands. Note that pinching is not a necessary condition for gesture articulation.

the user is manually controlling an end effector. The requirement usually arises when the user needs to move the end effector along its axes (e.g. feeding a solder wire) or planes (e.g. changing a solder point). We selected a flapping elbow action as a toggle gesture to allow the user to switch between frames while working with both hands (Fig. 9). This gesture is also based on the results from [34].



(a) (b) (c) (d)

**Fig. 9** The toggle gestures for switching working frames can be performed by flapping the elbows. Figure (d) shows the detected flapping elbow motion in the UI.

Detailed discussion about how to recognize the gestures is described in the next section.

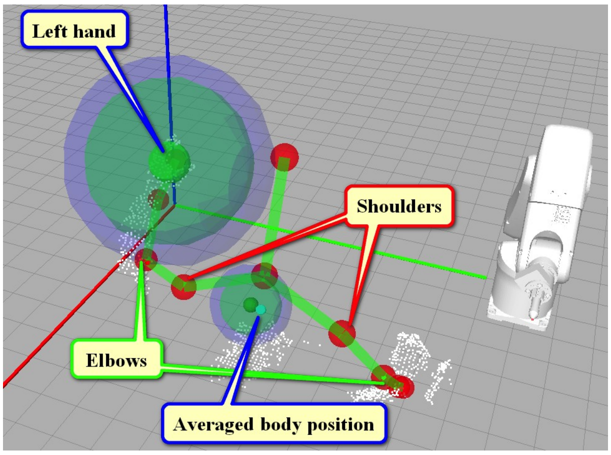


**Fig. 10** The upper body skeleton detected with Microsoft Kinect SDK (red dots connected with green lines). The shoulder joints are averaged as the body position. The elbows joints are used to detect flapping elbow gestures.

### 4.2 Gesture Recognition

In this study we propose a rubber band model for implementing body movement and hand gesture recognition. The model allows the users to start and stop controlling an end effector at any point in the workspace. Gesture recognition states and information such as hand positions and a recognized user skeleton are displayed in the rviz display (Fig. 10).

The proposed model can be visualized using the metaphor of tying an object (e.g. a user's hands) to a pivot point with a rubber band. When the user moves the hand inside the workspace, an initial position (the smallest circle in Fig. 11(a)) moves with the hand until it is held for a certain time for initialization (a green circle at the left hand in Fig. 10). After initialization the position is fixed as a pivot point for gesturing. At this stage, the user can articulate gestures to control the end effector within the area between the middle and large circles (Fig. 11(b)). If the user wants to stop controlling, he/she can either move the hand back to the pivot point (Fig. 11(a)) or move the hand outside of the large circle (Fig. 11(c)).

In other words, at the initial state (Fig. 11(a)) the rubber band is not stretched enough to enable gesture control. This allows a gesture recognizer to deal with a noisy position measurement and unintended initialization. When the hand is moved further from the pivot point (Fig. 11(b)), the direction and length of the rubber band can be used to control direction and velocity. The rubber band will rupture and a replacement (reinitialization) is needed if the hand is moved too far from the pivot point (Fig. 11(c)). A certain initialization time is needed before starting to control the end

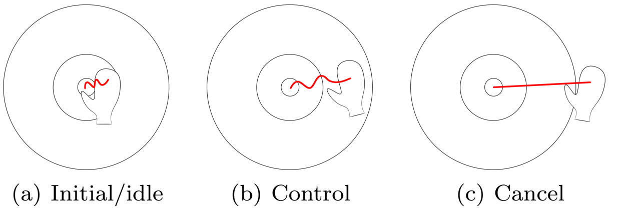


**Fig. 11** Three states of the rubber band model. The red string is for visualizing the rubber band.

effector to ensure that the manual control is intentionally activated.

### *4.2.1 Body Movement Gestures*

Body movement gestures are recognized using skeleton-detecting functions in Windows Kinect SDK. The detected skeletons are filtered and smoothed before being sent to be displayed and processed in the main PC. In this study, we utilize only wrists, elbows, shoulders, neck, and head joint information as shown with small red spheres in Fig. 10.

The left and right shoulders of the skeleton are averaged as a reference point for body movement recognition as shown in Fig 10. The average position is used because both joints are stable for detecting upper body motions with the SDK version 1.5. The spheres between the left and right shoulders in Fig. 10 are the visualization of the rubber band model and are used as feedback information for users.

When the participants tilt, lean, or twist their bodies, the averaged shoulder position will move from its initial position (replace hand in Fig. 11 with the averaged position of the shoulder at a neutral seating position). The displacement and direction of the averaged position is used for computing the moving direction and speed of the robot end effector.

The gesture of toggling between workspace and end effector frames is recognized by detecting a flapping movement of the elbows (Fig. 9). The detection is based on a one-shot state machine that uses elbows joints displacement and direction as show in Fig. 9(d) as it inputs. A completed cycle of the elbow joint (up and down) is needed for triggering the state machine.

### *4.2.2 Hand Gestures*

Hand gestures are recognized by functions in the PCL. Arm-like point cloud clusters are classified using principal component analysis (PCA) function by searching for elongated objects floating above the desk. All points that belong to the structure of the robot are filtered out with occupancy map monitor functions in the MoveIt library to make the arm-like point cloud cluster easier to detect. Hand positions are computed from clusters of the point cloud near the end of the arm-like cluster as shown in Fig. 12.

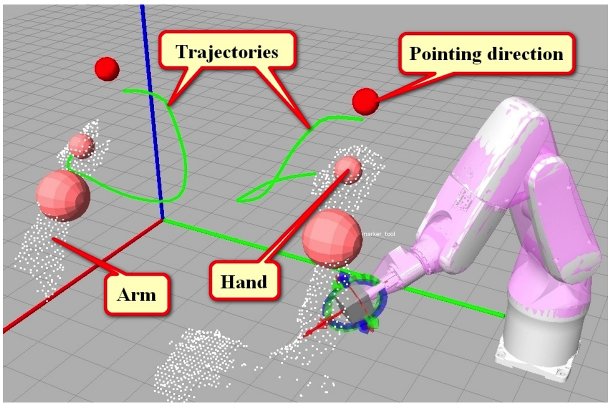


**Fig. 12** Hand detection with PCL. The white dots are point cloud clusters of all the objects in the workspace. Hand positions are the small pink spheres. Hand trajectories are displayed with green lines.

Hand positions are smoothed by the Kalman filter functions from OpenCV library [7]. The filter smooths positions and velocities ($x_k = [x, y, z, v_x, v_y, v_z]^T$) of the hands by using a state transition matrix ($F_x$) as shown in Eq. 1. The d$t$ was set to 30 Hz according to frame rate of the cameras. Inputs for the measurement update are centroid $(x, y, z)$ of point clouds of each hand as shown in Fig. 12.

$$F_x = \begin{bmatrix} 1 & 0 & 0 & \mathrm{d}t & 0 & 0 \\ 0 & 1 & 0 & 0 & \mathrm{d}t & 0 \\ 0 & 0 & 1 & 0 & 0 & \mathrm{d}t \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

The filters are tuned to ensure a balance between the smoothness and responsiveness of detected hands. Additional information about Kalman filter can be found in [32]. Furthermore, we also implemented a state machine to track, update, correct, and reset state of the filters when the participant is moving hand(s) inside, into or from the working area.

Using the same rubber band model, spheres around the left hand in Fig. 10 are a visualization of the model and are used as feedback information for the users while they are interacting with the robot. The recognizer interprets one- and two-hand gestures for translation (Fig. 7) and orientation control (Fig. 8), respectively.

## 5 Experiment

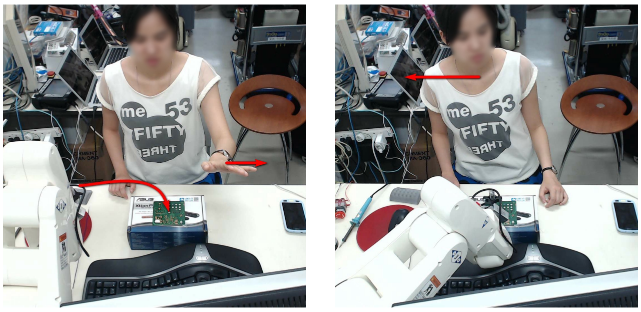
### 5.1 Introduction and Participants

The experiment focuses on usability testing to validate the proposed gestures with the helping hand robot. We set up a soldering task to compare the proposed gestures (hand and body movement gestures) with a 3D UI (the interactive marker in Fig. 3(b)) as shown in Fig. 4.

Eight participants, all students of Keio University, volunteered for the experiment. Three of them were women. The average age of the participants was 26.3, $SD = 2.2$. All participants were familiar with computer systems but had no experience with an industrial robot. They had experience with a manual soldering tasks before the experiment. All participants had experience with 3D games or 3D CAD software and had 3D gesture control experience with a modern game console such as Wii, Xbox, or PlayStation.

### 5.2 Experimental Procedure

The experiment began with an explanation of the purpose of the study before a demonstration of the proposed gestures and system usage by an instructor. The instructor demonstrated how to manually control an end effector with body movement (Fig. 1), hand movement (Fig. 7 and 8), by flapping elbows (Fig. 9), and by using the interactive marker (see Section 3.2). After the demonstration, the participants practiced the use of all the gestures and the interactive marker to ensure that they knew how to control the helping hand robot manually using all methods (Fig. 13).

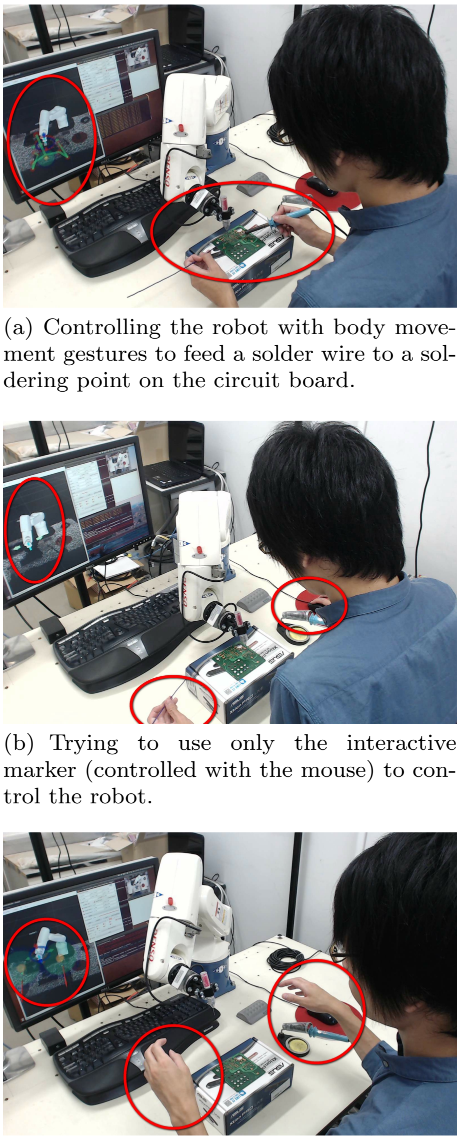


(a)

(b)

**Fig. 13** (a) The participant tries to set up the end effector with hand gestures and then (b) tries to control the robot with body movement gestures.

After finishing all the practice runs, the participants were asked to perform a simulated soldering task with an unplugged soldering iron (for safety reasons). The task had two steps, as mentioned in Section 4.1. The first step was to set up an end effector by manually controlling it from its start position to an area above the circuit board (Fig. 13(a)). In the second step, the participants were asked to try to solder a cable to three specified points on the circuit board with help from the robot (Fig. 14).



(a) Controlling the robot with body movement gestures to feed a solder wire to a soldering point on the circuit board.

(b) Trying to use only the interactive marker (controlled with the mouse) to control the robot.

(c) Trying to set up the orientation of the end effector with hand gestures.

**Fig. 14** The experiment showed the advantage of the body movement gestures in that the participant can hold a tool and an object in the working pose while engaging with the task (a) without having to switch back and forth when compared with other methods (b and c).

The participants were not explicitly asked to hold the soldering iron and the cable in their hands before performing the experiment; this was to observe how the participants grasped and released objects in three soldering task trials using the interactive marker, hand gestures, and body movement gestures. The order of the experiment was not randomized because all participants already practiced all controlling methods (hand, body, and interactive marker) under supervision of the authors before participating in the experiment.

After the experiments, the participants answered a questionnaire and discussed their opinions and suggestions for the system with the instructor. The instructor took notes during the experiment and all sessions were video recorded for further analysis.

### 5.3 Metrics

The participants were asked to rate the tasks using a set of seven-point Likert scales (1 – disagree to 7 – agree) and to answer a number of demographic questions after the experiment. The Likert scales begin with three pairs of scales for measuring an opinion about the proposed gestures. The scales can be read as "x gestures are suitable for the purpose" and "x gestures are easy to remember and use" where "x" are "hand", "body movement", and "elbow". The purposes of hand, body, and elbow movements are translation and orientation control, translation control, and working frame toggling, respectively.

The questionnaire continues with six additional Likert scales for comparing the use of the hand and body movement gestures with the interactive marker in both steps of manual control. The scales were divided into two groups that read "x is suitable for manually controlling the robot from the start position" (step 1) and "x is suitable for manually controlling the robot during the soldering task" (step 2) where "x" are "hand gestures", "body movement gesture", and "the interactive marker".

The participants were asked if "it is acceptable to change the method for controlling the robot during the task", for example, switching between hand and body movement gestures and the interactive marker as they see fit.

### 5.4 Statistical Results

The average score from Likert scales indicated that the hand $((M = 5.6,\ SE = 0.32), (M = 5.8,\ SE = 0.31)$, body movement $((M = 5.9,\ SE = 0.40), (M = 5.8,\ SE = 0.49))$ , and flapping elbow gestures $((M = 6.1,\ SE = 0.35), (M = 6.1,\ SE = 0.40))$ are suitable for their purposes and can be remembered and used without difficulty. The average scores and standard error bars are shown in Fig. 15.

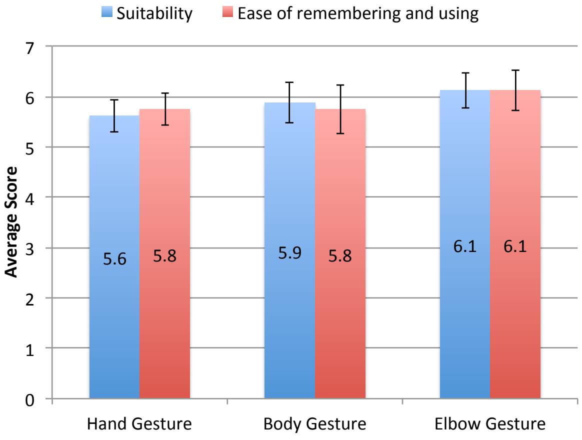


**Fig. 15** The average score of suitability and ease of remembering and using the hand, body movement, and flapping elbow gestures.

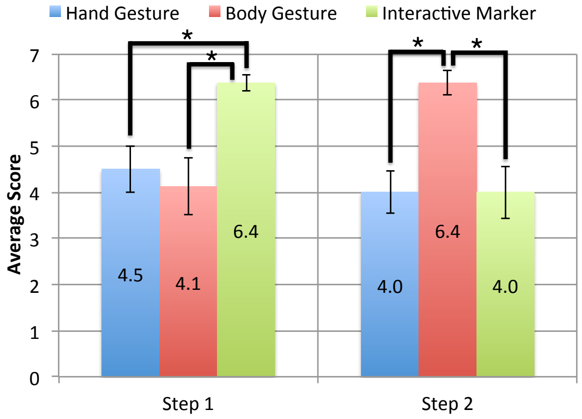


**Fig. 16** Average scores of the suitability of the proposed gestures and the interactive marker in the first and second steps of the manual control experiment.

Because each participant performed all manual control methods, we conducted a one-way within-subjects ANOVA to compare the preferences of the participants regarding manual control in the first and second steps. The average scores of the first and second steps of the manual control are shown in Fig. 16 and differences between the two steps were found to be statistically significant at the $p < 0.05$. Post hoc analysis adjustments are based on the Bonferroni method.

For the first step, there was a significant difference between the control methods, $F(2, 14) = 9.00,\ p < 0.05$. The post hoc analyses (Table 1) indicated that the interactive marker $(M = 6.4,\ SE = 0.18)$ was pre-

**Table 1** T-test results of the first step. The (*) indicates that the difference is significant.

| Pair | Result |
|---|---|
| Hand & Body | $t(7) = 0.75,\ p = .24, (\beta - 1) = 0.15$ |
| Hand & Marker* | $t(7) = -3.91,\ p < .01, (\beta - 1) = 0.99$ |
| Body & Marker* | $t(7) = -3.21,\ p < .01, (\beta - 1) = 0.98$ |

**Table 2** T-test results of the second step. The (*) indicates that the difference is significant.

| Pair | Result |
|---|---|
| Hand & Body* | $t(7) = -5.16,\ p < .01, (\beta - 1) = 1.00$ |
| Hand & Marker | $t(7) = 0.00,\ p = .50, (\beta - 1) = 0.05$ |
| Body & Marker* | $t(7) = 3.64,\ p < .01, (\beta - 1) = 1.00$ |

ferred over the hand ($M = 4.5$, $SE = 0.50$) and body movement ($M = 4.1$, $SE = 0.61$) gestures with statistical power $(\beta - 1)$ greater than 0.8. Hand gestures were slightly more preferred over body movement gestures, but the differences were not statistically significant in the first step of manual control with power less than 0.2. The statistical power or $(\beta - 1)$ is used to determine the type II error rejection of the test. Normally, $(\beta - 1 < 0.2)$ is too weak and $(\beta - 1 > 0.8)$ is strong enough for validating the study.

For the second step, there was a significant difference ($F(2, 14) = 10.93$, $p < 0.05$) between body movement gestures ($M = 6.4$, $SE = 0.26$), hand gestures ($M = 4.0$, $SE = 0.46$), and the interactive marker ($M = 4.0$, $SE = 0.56$). The post hoc analysis (Table 2) showed that body movement gestures were preferred over hand gestures and the interactive marker with significant differences and have statistical power greater than 0.8. There was no significant difference between the hand and interactive marker in the second step with statistical power less than 0.2.

The statistical power computation is based on a post hoc power analysis that computes archived power using mean and standard deviation of each pair of the experiment [10].

Furthermore, the participants also showed that they were willing to switch between control methods if it helped complete the task and made their work easier ($M = 5.8$, $SE = 0.59$).

## 6 Discussion

### 6.1 Results from the Experiment

In summary, the participants were able to manually control the end effector with the proposed gestures without noticeable difficulty. The results showed that the body movement gestures were preferred in the second step, whereas the interactive marker was preferred over gestures for setting up the end effector in the first step.

The behavioral observation shows that the participants have two distinguishable ways of holding and releasing objects during the experiment. For the interactive marker and hand gestures, all participants grasped and held the soldering iron and the cable only when they were performing the soldering task on each soldering point. The participants released the objects (putting them on the table) immediately before starting to control the movement of the robot to the next soldering point. For the body movement gestures, after picking up the soldering iron and the cable for the first soldering point, only one participant released the soldering iron and the cable before starting to control the robot to the next soldering point. These findings emphasize that body movement gestures can be useful for manual tasks when there is a need to continuously hold tools and objects.

The participants also commented during an interview that it would make more sense for the first step to be performed automatically by the robot and it is acceptable if fine-tuning is needed. This qualitative data informs that the participants expected the robot to move automatically when the robot have to move in a large distance. However, gestures are acceptable when the robot is struggling in complicated situations. This comment supports the use of gestures in the example scenario in Fig. 2, which addressed a glitch in the interactive function of the robot system.

Although hand gestures showed no significant difference between the first and second steps, the average score of the gestures and the participants' comments still encourage the use of hand gestures as a supplementary or alternative control method when other methods, such as the interactive marker or body movement gestures, are not appropriate.

### 6.2 User Preferences

The proposed gestures utilize the rubber band model (Fig. 11) for gesture state recognition. The distance from the initial position for idle, control, and cancel states must be specified. From observation of and discussions with the participants, we found that the participants exhibited noticeable preferences over the predefined distances. Many participants preferred small and precise displacement control, while others requested large and explicit hand and body movements.

A conclusion from the discussions with the participants also shows that the main reason that the interactive marker was preferred over both types of gestures in the first step was the moving speed of the end effector (it was much slower when controlled with gestures for safety reasons); this, in fact, can be altered to match each participant's preferences. However, additional effort for the safety and reliability of gesture recognition will be necessary.

### 6.3 User-defined Gestures

All selected gestures were based on a user-centered design approach. The gestures were derived from a previous study [34] in which an experiment was conducted using a virtual robot. The present study implemented both the helping hand robot and gesture recognition to realize user-defined gestures for manual control of an end effector of the helping hand robot. With the intuitive gestures, users can start and stop controlling the end effector whenever they need to. The body movement gestures allow the users to control the end effector with high precision while both hands are engaged with tasks. Without restriction on hand and initial positions, users can articulate one- or two-handed gestures while holding objects in their hands. The implementation also allows users to articulate dexterous body and hand gestures without additional devices such as gloves (e.g. [27]) or sensing devices (e.g. [19]).

### 6.4 System Implementation

The current system is designed based on ease of implementation and flexibility for the experiment. It uses mostly off-the-shelf software and hardware. A more specific and efficient software implementation should be able to help reducing the number of devices.

### 6.5 Limitations and Future Work

This study described the detailed implementation of a set of user-defined gestures for manual control of helping hand robots. The experiment showed that users could freely control the end effectors of a helping hand robot with the proposed gestures.

However, as suggested by a number of participants, an additional UI that would allow them to know the current state of the system, such as robot joint limits or the state of gesture recognition, without looking at the screen would enhance the efficiency of the system. This suggestion implies that the implemented UI (Fig. 6) might influence how the participants use gestures to control the robot. Although we expect that the proposed gestures for manual control are natural for the participants, additional studies of the system without a traditional UI or a robot that can elicit gestures (e.g. [9]) are needed.

User preferences in Section 6.2 emphasize the need of customization functions for various aspects of the system. From the implemented system point of view, these kinds of adjustments are tedious, and hence automatic calibration functions or setting methods will surely enhance the system efficiency and user experience. Rubber band model and robot moving speed are also the important topics that should be able to customize by users.

The current implementation was limited to a robot that was mounted on a table and facing its user. Different robots and configurations, such as a robot that is mounted on a linear unit for extending the working envelope, a mobile manipulator robot, or a robot working side-by-side with the user, will require additional gestures and sensing effort to handle the additional DOFs and the variety of user positions with respect to the robot.

Switching between control methods was not a significant burden, as indicated by the results in Section 5.4, and hence multimodal manual control for assisting or setting up a helping hand robot with various methods, such as gestures and 3D user interfaces, could be more useful than using just one particular method. However, additional effort and further studies would be required to confirm this.

Gestures or interaction methods that allow users to control the trajectory of the robot will open a new perspective for use of the helping hand robots. Industrial robots have already been used in various art-related domains such as cinematography, architecture, and installation arts. In such domains, expressing one's creativity, in activities such as drawing robot trajectories or setting camera direction through direct interaction with a robot using natural gestures might be more intuitive than tedious work with mouse clicks through a 3D virtual world in traditional UI.

The hand and body movements gestures could help specific handicapped persons such as deaf or semi-paralysis to interact with robots or machines easier. Furthermore, rehabilitation such as a process for recovering motor skills (e.g. hand/arm movement) after injury could also benefit from robot motions if the robot could sense and move according to quality of patient motions (the analogy of user-defined gestures). This applications would require additional experimental trials for validating and adjusting before clinical testing.

## 7 Conclusion

We have presented an implementation of a helping hand robot system that can be manually controlled with a set of user-defined gestures that were derived from a previous study. The implemented system and selected gestures allow users to control an end effector while working closely with the robot using body movement and hand gestures. The gesture-recognition module allows the user to start and stop controlling at any position within the workspace. In particular, the users were able to control the helping hand robot with body movement gestures even though both their hands were occupied with the task. In addition, we conducted an experiment to confirm the benefit of our proposed system. The results show that the proposed gestures can be useful as a complementary feature for the development of multimodal communication in HRI and HRC to make the helping hand robot interact more naturally with humans.

## References

1. 3D visualization tool for ROS (Accessed July, 2013). URL http://wiki.ros.org/rviz
2. HiveGround ROS Packages (Accessed July, 2013). URL https://github.com/hiveground-ros-package
3. Kinect SDK (Accessed July, 2013). URL http://www.microsoft.com/en-us/kinectforwindows
4. MoveIt! (Accessed July, 2013). URL http://moveit.ros.org/wiki/MoveIt!
5. Spacenavigator (Accessed July, 2013). URL http://www.3dconnexion.com/
6. Bauer, A., Wollherr, D., Buss, M.: Human-robot Collaboration: A Survey. Humanoid Robotics **05**(01), 47–66 (2008)
7. Bradski, G., Kaehler, A.: Learning OpenCV. O'Reilly Media (2008)
8. Burger, B., Ferrane, I., Lerasle, F., Infantes, G.: Two-handed gesture recognition and fusion with speech to command a robot. Autonomous Robot **32**(2), 129–147 (2012)
9. Ende, T., Haddadin, S., Parusel, S., Wusthoff, T., Hassenzahl, M., Albu-Schaffer, A.: A human-centered approach to robot gesture based communication within collaborative working processes. In: Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 3367–3374 (2011)
10. Faul, F., Erdfelder, E., Lang, A.G., Buchner, A.: G*Power 3: A flexible statistical power analysis program for the social, behavioral, and biomedical sciences. Behavior Research Methods **39**(2), 175–191 (2007)
11. Gleeson, B., MacLean, K., Haddadi, A., Croft, E., Alcazar, J.: Gestures for industry: intuitive human-robot communication from human observation. In: Proc. of the 8th ACM/IEEE Int. Conf. on Human-robot interaction, pp. 349–356 (2013)
12. Goodrich, M.A., Schultz, A.C.: Human-robot interaction: a survey. Foundation and Trends in Human-Computer Interaction **1**, 203–275 (2007)
13. Gossow, D., Leeper, A., Hershberger, D., Ciocarlie, M.: Interactive Markers: 3-D User Interfaces for ROS Applications [ROS Topics]. IEEE Robotics Automation Magazine **18**(4), 14–15 (2011)
14. Groothuis, S., Stramigioli, S., Carloni, R.: Lending a Helping Hand: Toward Novel Assistive Robotic Arms. IEEE Robotics Automation Magazine **20**(1), 20–29 (2013)
15. Hammel, J., Hall, K., Lees, D., Leifer, L., der Loos, M.V., Perkash, I., Crigler, R.: Clinical evaluation of a desktop robotic assistant. Rehabilitation Research and Development **26**, 1–16 (1989)
16. Hato, Y., Satake, S., Kanda, T., Imai, M., Hagita, N.: Pointing to space: modeling of deictic interaction referring to regions. In: Proc. of the 5th ACM/IEEE Int. Conf. on Human-robot interaction, pp. 301–308 (2010)
17. Iossifidis, I., Theis, C., Grote, C., Faubel, C., Schoner, G.: Anthropomorphism as a pervasive design concept for a robotic assistant. In: Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, vol. 3, pp. 3465–3472 (2003)
18. Kuniyoshi, Y., Inaba, M., Inoue, H.: Learning by watching: extracting reusable task knowledge from visual observation of human performance. IEEE Trans. on Robotics and Automation **10**(6), 799–822 (1994)
19. Neto, P., Pires, J., Moreira, A.: Accelerometer-based control of an industrial robotic arm. In: Proc. of . IEEE Int Symp.on Robot and Human Interactive Communication, pp. 1192–1197 (2009)
20. Nof, S.Y. (ed.): Handbook of Industrial Robotics, second edn. John Wiley & Sons (1999)
21. Oikonomidis, I., Kyriazis, N., Argyros, A.: Tracking the articulated motion of two strongly interacting hands. In: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, pp. 1862 –1869 (2012)
22. Oviatt, S.: Ten myths of multimodal interaction. Communication of the ACM **42**(11), 74–81 (1999)
23. Quigley, M., Conley, K., Gerkey, B.P., Faust, J., Foote, T., Leibs, J., Wheeler, R., Ng, A.Y.: ROS: an open-source Robot Operating System. In: ICRA Workshop on Open Source Software (2009)
24. Rogalla, O., Ehrenmann, M., Zollner, R., Becher, R., Dillmann, R.: Using gesture and speech control for commanding a robot assistant. In: Proc. of IEEE Int. Workshop on Robot and Human Interactive Communication, pp. 454–459 (2002)
25. Rusu, R.B., Cousins, S.: 3D is here: Point Cloud Library (PCL). In: Proc. IEEE Int. Conf. Robotics and Automation, pp. 1–4 (2011)
26. Stiefelhagen, R., Fugen, C., Gieselmann, R., Holzapfel, H., Nickel, K., Waibel, A.: Natural human-robot interaction using speech, head pose and gestures. In: Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, vol. 3, pp. 2422–2427 (2004)
27. Tran, N., Phan, H., Dinh, V., Ellen, J., Berg, B., Lum, J., Alcantara, E., Bruch, M., Ceruti, M., Kao, C., Garcia, D., Fugate, S., Duffy, L.: Wireless Data Glove for Gesture-Based Robotic Control. In: J. Jacko (ed.) Human-Computer Interaction. Novel Interaction Methods and Techniques, *Lecture Notes in Computer Science*, vol. 5611, pp. 271–280. Springer Berlin / Heidelberg (2009)
28. Wachs, J., Stern, H., Edan, Y.: Cluster labeling and parameter estimation for the automated setup of a hand-gesture recognition system. IEEE Trans. on Systems, Man and Cybernetics, Part A: Systems and Humans **35**(6), 932–944 (2005)

29. Wachs, J.P., Kölsch, M., Stern, H., Edan, Y.: Vision-based hand-gesture applications. Communications of the ACM **54**, 60–71 (2011)
30. Wallhoff, F., Blume, J., Bannat, A., Rsel, W., Lenz, C., Knoll, A.: A skill-based approach towards hybrid assembly. Advanced Engineering Informatics **24**(3), 329–339 (2010)
31. Wang, R., Paris, S., Popović, J.: 6D hands: markerless hand-tracking for computer aided design. In: Proc. of the 24$^{\mathrm{th}}$ annual ACM symp. on User interface software and technology, UIST '11, pp. 549–558. ACM (2011)
32. Welch, G., Bishop, G.: An introduction to the Kalman Filter. Tech. rep., Department of Computer Science University of North Carolina at Chapel Hill Chapel Hill (2006)
33. Wobbrock, J.O., Morris, M.R., Wilson, A.D.: User-defined gestures for surface computing. In: Proc. of the ACM Int. Conf. on Human Factors in Computing Systems, pp. 1083–1092 (2009)
34. Wongphati, M., Kanai, Y., Osawa, H., Imai, M.: Give Me a Hand How Users Ask a Robotic Arm for Help with Gestures. In: Proc. of IEEE Int. Conf. on Cyber Technology in Automation, Control, and Intelligent Systems, pp. 64 – 68 (2012)

## 8 Biographies

**Mahisorn Wongphati** is a Ph.D. student of Graduate School of Science and Technology at Keio University. He received his M.Eng. and B.Eng. in Computer Engineering from Faculty of Engineering, Chulalongkorn University, Thailand. His research interests include mobile robots, human-robot interaction, human-robot collaboration, image processing, robot system development, and artificial intelligent. Since October 2013, he is working with a robotic company he co-founded called Hive-Ground Co.,Ltd. in Thailand.

**Hirotaka Osawa** is an assistant professor in University of Tsukuba. His research field is in human-agent interaction, particularlyanthropomorphization of an object. His own research focuses on how human-like appearance and attitude improves interaction between a user and machines. He wants to create universal users interface experiences using our innate response to the world. His research interest is an improvement of todays complex household appliances. Dr. Osawa received his Ph.D. in Engineering, M.S. and B.S. in Computer Science from Keio University.

**Michita Imai** is a Professor of Faculty of science and technology at Keio university and a Researcher at ATR Intelligent Robot Laboratories. He received his Ph.D. degree in Computer Science from Keio Univ. in 2002. In 1994, he joined NTT Human Interface Laboratories. He joined the ATR Media Integration & Communications Research Laboratories in 1997. He was a visiting scholar ofUniversity of Chicago from 2009-2010. His research interests include autonomous robots, human-robot interaction, speech dialoguesystems, humanoid, and spontaneous behaviors. He is a member of Information and Communication Engineers Japan (IEICE-J), theInformation Processing Society of Japan, the Japanese Cognitive Science Society, the Japanese Society for Artificial Intelligence, Human Interface Society, IEEE, and ACM.