Development of a smart sleeve control mechanism for active assisted living

Cheng, Alexander Liu; Santos, Caio; Santos, Pedro; Vega, Nestor Llorca

DOI
10.1109/WF-IoT.2019.8767314

Publication date
2019

Document Version
Final published version

Published in
Proceedings of the IEEE 5th World Forum on Internet of Things (WF-IoT)

Development of a Smart Sleeve Control Mechanism for Active Assisted Living

Alexander Liu Cheng[1,2], Caio Santos[3], Pedro Santos[4], Nestor Llorca Vega[2,5]
[1]Faculty of Architecture and the Built Environment, Delft University of Technology, Delft, The Netherlands
[2]Facultad de Arquitectura e Ingenierías, Universidad Internacional SEK, Quito, Ecuador
[3]Department of Nuclear Engineering, Federal University of Rio de Janeiro, Rio de Janeiro, Brazil
[4]Department of Electronic and Computer Engineering, Federal University of Rio de Janeiro, Rio de Janeiro, Brazil
[5]Escuela de Arquitectura, Universidad de Alcalá, Madrid, Spain

E-mail: a.liucheng@tudelft.nl, caiofmsantos@poli.ufrj.br, pedrocms@poli.ufrj.br, nestor.llorca.arq@uisek.edu.ec

Abstract*—This paper describes the development of a Smart Sleeve control mechanism for Active Assisted Living. The Smart Sleeve is a physically worn sleeve that extends the user's control capability within his/her intelligent built-environment by enabling (1) direct actuation and/or (2) teleoperation via a virtual interface. With respect to the first, the user may point his/her sleeve-wearing arm towards a door or a window and actuate an opening or a shutting; or towards a light and effect its turning on and off as well as regulating its intensity; or even towards a particular region to initiate ventilation of it. With respect to the second, the user may engage actuations in systems beyond his/her field of vision by interacting with a virtual representation of the intelligent built-environment projected on any screen or surface. For example, the user is able to shut the kitchen's door from his/her bedroom by extending the Smart Sleeve towards the door of a virtual representation of the kitchen projected (via a standard projector and/or television monitor) on the bedroom's wall. In both cases, the Smart Sleeve recognizes the object which the user wishes to engage with—whether in the real or the virtual worlds—(a) by detecting the orientation of the extended arm via an Accelerometer / Gyroscope / Magnetometer sensor; and (b) by detecting a specific forearm muscle contraction via Electromyography caused by the closing and opening of the fist, which serves to select the object. Once the object is identified and selected with said muscle contraction, subsequent arm gestures effect a variety of possible actuations for a given object (e.g., a window may be opened / shut or dragged to differing degrees or aperture; a light's intensity may be increased or decreased, etc.). In cases of ambiguous selections, the user my use voice-commands to explicitly identify the desired object of selection. Furthermore, due to this voice-based recognition mechanism, which recognizes both spoken commands as well the identity of the speakers, different rights to actuation may be assigned to different users. The Smart Sleeve is yet another mechanism integrated into an on-going development of a highly intuitive Active and Assisted Living implementation and its features, ones explicitly designed to enhance user-experience as well as to promote user well-being via intuitive interactions between human and non-human agents within the intelligent built-environment.

***Keywords*—Remote Control, Teleoperation, Design-to-Robotic-Operation, Intelligent Built-environments, Wireless Sensor and Actuator Networks**

I. INTRODUCTION

The work detailed in this paper is situated within the intelligent built-environment discourse. It is part of an on-going development of increasingly sophisticated and intuitive systems that, while informed by architectural considerations, focus on computational intelligence and information services to enable intelligent built-environments. Such systems aim to extend the capabilities of existing *Ambient Intelligence* (AmI) [1] / *Ambient Assisted Living* or *Active and Assisted Living* (AAL) solutions, where the focus is primarily centered on *Information and Communication Technologies* (ICTs) (see [2] for example) without regard for the built-environment itself. By detaching ICT considerations from those involving the host within which they are embedded / integrated, an important difference in *Technology Readiness Levels* (TRLs) between ICT-based systems / services and the built-environment is invariably created. This difference may lead to a compromise in performance and/or a reduction in *Quality of Service* (QoS) with respect to said ICT-based systems / services, which decreases serviceable intelligence in the built-environment. This consequence is due to a lack of complementarity between the deployed technical technologies and architectural ones upon which the intelligent built-environment supervenes. In order for a system (e.g., the intelligent built-environment) to perform effectively, its constituent sub-systems (e.g., ICT-based mechanisms and services as well as architectural technologies) must be mutually complementary [3]. In this paper, a Smart Sleeve control mechanism is presented as an interaction interface between the user and his/her intelligent built-environment. Via this mechanism, the user may engage (1) directly with objects / actuating systems within his/her field of vision, or (2) indirectly from a different region within the same built-environment via a visual projection (on a monitor or projector) of said environment's virtual representation. The Smart Sleeve works with muscle contractions to select / deselect objects and with an Accelerometer / Gyroscope / Magnetometer sensor to translate subsequent arm gestures into actuating commands (e.g., open, shut, increase or decrease illumination, etc.) (see Section II for a more detailed explanation). The objective of this mechanism is to extend the user's reach and capabilities of engaging with his/her built-environment with minimal necessary effort, which may be particularly pertinent for users with limited mobility and/or reach. This mechanism is added into an existing catalog of highly intuitive AAL mechanisms and services, all designed to enhance the quality of life, spatial experience, and well-being of the user. Accordingly, the Smart Sleeve is built on a decentralized and scalable System Architecture (see [4] for a brief overview) and subsumed into its technical ecosystem.

II. Concept and Approach

The Smart Sleeve enables the user's arm to work as a pointer. The user selects / deselects objects and/or actuating systems in his/her built-environment by pointing towards them and contracting the forearm's muscles (via the opening and closing of the fist). Once an object / actuating system is selected, subsequent arm movements / gestures are translated into actuation commands acting on said object / actuating system. Some commands are binary (e.g., open or shut the door) while others graduate in degrees (e.g., increase or decrease illumination intensity). The selection of objects may be direct (inside the user's field of vision) by pointing at the actual object or indirect (outside the user's field of vision) by pointing at its virtual representation. Regardless of whether the engagement is direct or indirect, a virtual representation of the built-environment may be projected any time. This representation is optional when the engagement is direct but necessary when it is indirect. In this representation, the user's arm is illustrated as a black line with one end anchored at the center of the virtual world and the other moving with the orientation of the arm's pointing. The virtue of the Accelerometer / Gyroscope / Magnetometer sensor integrated in the Smart Sleeve (see Fig. 1), the orientation of the user's arm relative to the built-environment is consistently tracked and represented in the virtual world. When the user is engaging directly with an object, he/she simply points, closes and opens his/her fist to select it, and then waves up or down, left or right, depending on what such gestures are configured to mean for the object. However, when the user is engaging indirectly with an object, the control sequence is more elaborate. For example, imagine that the user is lying in bed in his/her bedroom and that he/she forgot to shut the kitchen's door. He/she would request the built-environment to show its virtual world, either on a monitor or via a projector, and the user would first indicate which region of the built-environment he/she would like to engage with. Upon selecting a particular space, the virtual world would show this space as if viewed from the center of a wall. That is, the representation of the space in the virtual world would not rotate to match the orientation of the user's arm while lying in bed—perhaps this orientation would be an awkward one with respect to the kitchen. Instead, this neutral point of view (see Fig. 2, Fig. 3, and Fig. 4) enables the user to correlate his arm's physical orientation—at the time of beckoning the projection of the virtual world—with a centered orientation in the virtual kitchen. That is to say, if the user's arm parallel to the floor and pointing North at the time of beckoning the virtual world, then that orientation is the neutral orientation. This neutral orientation would always be perpendicular to the user's view of the virtual space in question. If the user does not move his/her pointing arm, then this pointer would look like a dot in the virtual world (a line perpendicular to the viewer would have both its start- and end-points aligned). But as soon as he/she moved his/her pointing arm, the start-point would remain in the center of the virtual world while the end-point would displace in the direction and magnitude corresponding to the physical arm's direction and magnitude—i.e., from this point onward, any motion left / right, up / down, etc. would be represented faithfully in the virtual world. To summarize on this distinction: when engaging with objects directly, the arm's pointing is absolute, while when engaging with objects indirectly, its pointing is relative. But in both cases, spatial displacement of the arm / pointer is accurately represented.

When an object is selected and engaged with, whether directly or indirectly, the corresponding arm gestures that can elicit an actuation are determined in a way that reflects how the arm would move if and when engaging physically with that same object. For example, a user swings a physical door open, therefore the swinging motion is a gesture that the built-environment's underlying system would construe as applicable to the door. Conversely, a punching motion would not be one that effects any actuation on the door, as it is normally uncommon to punch a physical door open. Nevertheless, an array of motions and gestures may be configured to effect a variety of actuations upon all actuating systems, depending on the user's preference. In the present implementation, the user may lift / lower a sliding window (see Fig. 2) and swing a door open / shut (see Fig. 3). The only configured arm motion that does not correspond to a physical one that is commonly effected upon an object is the turning of an illumination fixture on and off, and of its corresponding regulation of intensity. In the physical world, these would be caused by hand gestures / motions (e.g., pressing a switch, twisting a knob), and the present Smart Sleeve relies only on arm gestures. Accordingly, at present, the light fixture turns on and off when selecting and deselecting it, respectively. When on, moving the pointing arm towards the left reduces its intensity, while moving it towards the right increases it (see Fig. 4). As indicated earlier, the main objective of this *proof-of-concept* implementation is to demonstrate the feasibility and functionality of the Smart Sleeve as an aid in interacting with the intelligent built-environment, in particular for users with limited mobility (e.g., the elderly, people recovering from illnesses and/or accidents, et al.) and/or reach (e.g., children, expecting mothers, et al.). Finally, the Smart Sleeve, being a mechanism within a larger technical ecosystem, is also integrated with a previously developed Speech and Voice-Command Recognition mechanism [5]. Accordingly, the user may verbally specify a selection in situations where there is ambiguity. For example, imagine that the user points at the door directly, but at that angle a desk actionable lamp is also in the line of sight. In such a case the user may explicitly speak the name / identifier of the object he/she wishes to select in order to disambiguate. Furthermore, via this voice-based recognition mechanism's ability to recognize both the spoken command and the identity of the speaker, different actuation rights may be assigned to different users. For example, a user may have the right to open all windows and doors, while another may be assigned accessibility rights to specific ones only. This feature addresses potential safety concerns, especially where children are involved.

III. Methodology and Implementation

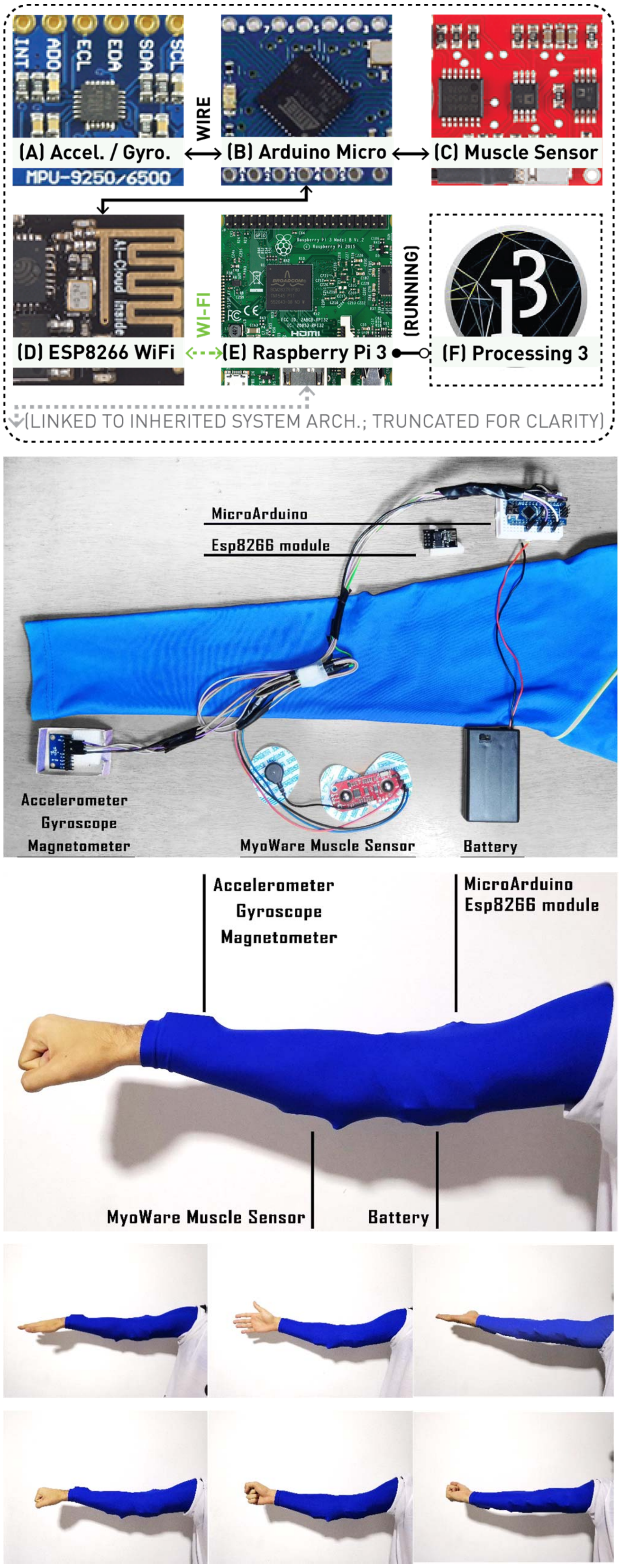

Fig. 1. System diagram and Smart Sleeve description (worn and unworn).

The *client* side of the Smart Sleeve mechanism is controlled by an Arduino / Genuino Mirco connected with an MPU-9250 Accelerometer / Gyroscope / Magnetometer and a MyoWare Muscle sensor. The MyoWare muscle sensor uses Electromyography (EMG) to detect forearm muscle contraction / release, which is correlated with the selection and deselection feature of the Smart Sleeve. The MPU-9250 sensor is used to detect the movement, velocity, and orientation of the Smart Sleeve-wearing arm. The incoming data is cleaned, processed (in a way based on [6]), and sent to the *server* side of the Smart Sleeve mechanism via an ESP8266 Wi-Fi module. On the *server* side, a Raspberry Pi 3 Model B (RPi3) that is part of the inherited System Architecture receives the data via the *Open Sound Control* (OSC) protocol and makes the stream available to a script in Processing 3.3 (see Fig. 1)—it is this script that also generates the virtualization of the intelligent built-environment (i.e., the virtual world). In a previous development, instead of the ESP8266 module and the OSC protocol, a 433MHz RF transmitter and corresponding receiver were used. However, the range of such a setup was too limited for the *client* side of the Smart Sleeve mechanism to interact with its *server* side at large distances.

Since the RRi3 *server* node is part of the larger System Architecture, it has access to all the ICT-based service mechanisms in the ecosystem. It is in this manner that the Smart Sleeve integrates with the Speech and Voice-Command Recognition mechanism developed via Google Cloud Platform®'s Cloud Speech-to-Text API [7]. As described elsewhere [5], this mechanism is capable of translating spoken speech to *String* text that can be used as triggering input for actuations. In this implementation, the user uses this mechanism by referring to the object it wants to select in cases of ambiguity. In order to do this, the service-triggering keyword (to initiate the service) must first be spoken to inform the system to translate what follows to *String* text. The user then must indicate that the commands that follow are directed to the Smart Sleeve mechanism (e.g., "For Smart Sleeve"). Finally, after receiving audio feedback that the specification is understood, the user proceeds to speak directed to the Smart Sleeve. For example, in the situation considered earlier, where the user points directly at the door yet encounters a desk lamp in the line of sight, the user must tell the system which of the available objects in the line of sight he/she wants to select (e.g., "the lamp", "the door"). Accordingly, the identified object is selected—and this selection is indicated in the virtual world by a change of color of the object in question; and in the real world via audio feedback. In order for this to work, the library of recognized words / commands in the Speech and Voice-Control Recognition mechanism must be current to include the catalog of actuating systems within the intelligent built-environment. It should be noted that for clarity, the built-environment within which the Smart Sleeve was implemented and tested was kept architecturally simple. The elements created include a basic door, sliding window, sliding floormat, central illumination, and a spherical geometry representing a non-descript piece of furniture.

IV. Results and Conclusions

As stated at the end of the last section, the simplified built-environment is a rectangular room that contains a basic door, sliding window, sliding floormat, central illumination, and a spherical geometry representing a non-descript piece of furniture. All of these elements are actionable, and in the present implementation all of them were engaged with several times to gauge precision, performance, and functionality. However, in this paper, figures corresponding to only three of these elements are illustrated. In the case of the window, the Smart Sleeve is pointed towards it, selects it, and an upwards movement is effected once the object is locked (see Fig. 2). Similarly, the case of the door, it is pointed at, selected, and swung open with the same gesture that is typically used to open hinged doors (see Fig. 3). Finally, in the case of the central light fixture, it is pointed at, selected, and gradually turned brighter or darker depending on whether the Smart Sleeve is oriented towards the right or left side of the room (see Fig. 4)

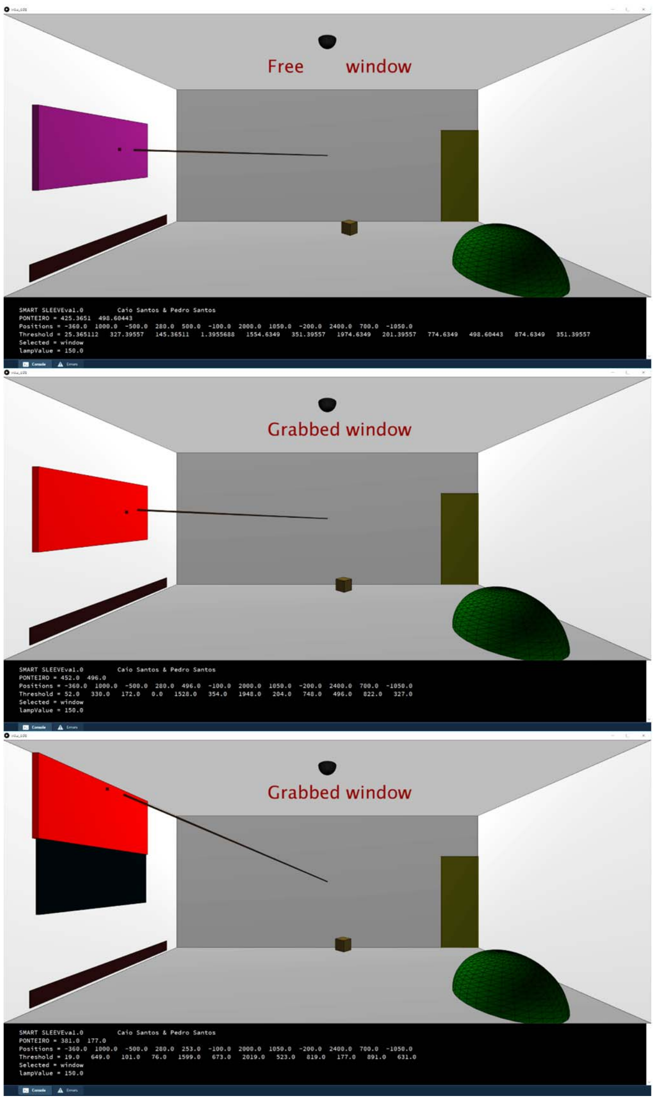

Fig. 2. Pointing (top), selecting (middle), and moving (bottom) sliding window.

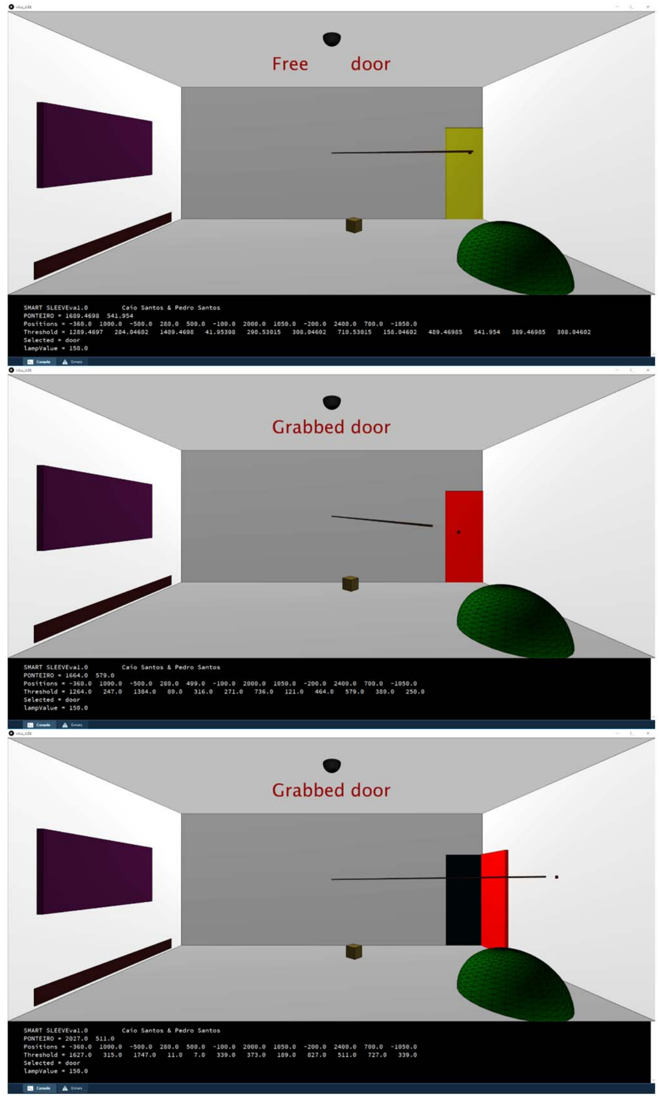

Fig. 3. Pointing (top), selecting (middle), and swinging (bottom) hinged door.

In all trials the Smart Sleeve performed successfully, which is promising yet unsurprising due to the controlled environment within which it was tested. Although the components of the Smart Sleeve are at a *Technology Readiness Level* (TRLs) [8] of 9, the solution itself must be crafted in a more intuitive and non-intrusive way before it may be considered as a feasible and comfortable device. Furthermore, cases of ambiguity were only tested with two overlapping objects (e.g., the door and the desk lamp in the same line of sight). Consequently, further tests with more complex scenarios must be tested in order to gauge the actual feasibility and TRL of the device. As the Smart Sleeve is optimized and further polished, it may be integrated with other existing mechanisms supervening on the inherited System Architecture. For example, further work is being conducted to integrate previously developed Object and Facial-Identity and -Expression Recognition mechanisms [5] based on *TensorFlow™* [9] and Google Cloud Platform®'s Cloud Vision API [10] with the Smart Sleeve's functionality.

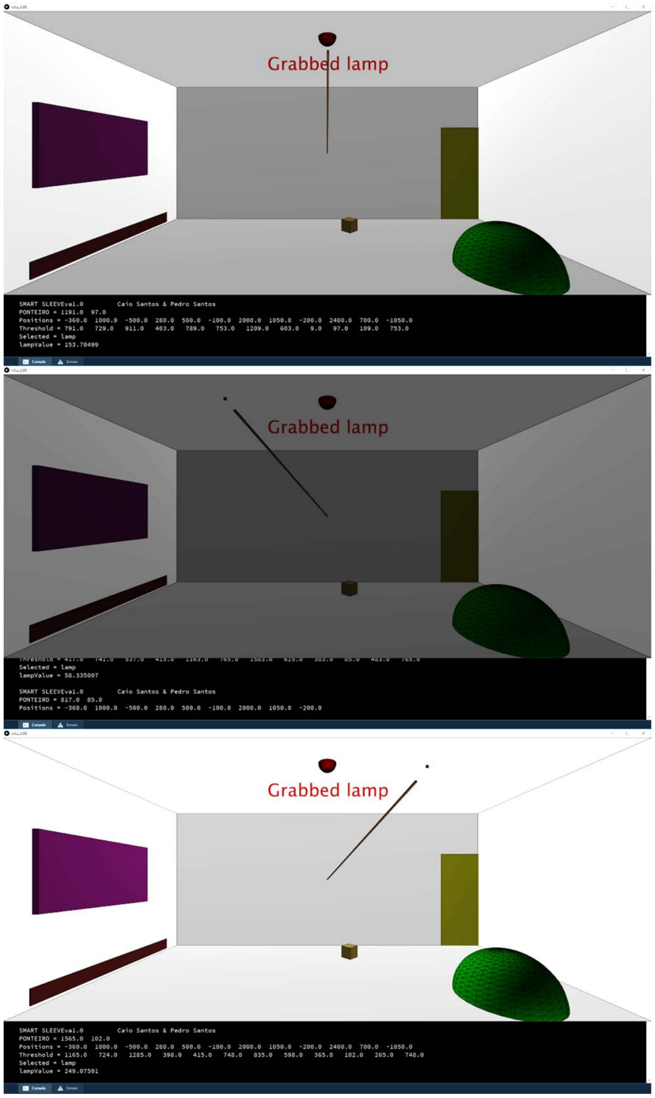

Fig. 4. From top to bottom: pointing, selecting, decreasing intensity (gesture left), and increasing intensity (gesture right) of central illumination device.

Further work is presently being undertaken to connect the Smart Sleeve mechanism with a robotic arm and gripper developed in parallel by the authors [11]. This robotic arm would be integrated into the intelligent built-environment in order to enhance a user's reach. In the present implementation the actuation devices are all integrated into the built-environment's objects themselves (e.g., actuating doors, windows, etc.), but the robotic arm would be a free agent capable of engaging with non-actuating objects (e.g., picking up objects remotely, etc.). One of the purposes of this future extension is to enable a user to lift objects that would otherwise be too heavy to lift. To be sure, the serviceability of this robotic arm is limited to particular areas within the built-environment at present, and key regions are being identified in order to increase the utility of this robotic extension. Finally, the Smart Sleeve, as a mechanism and solution, is presently being patented.

ACKNOWLEDGEMENT

An initial implementation of the work detailed in this paper was carried out at the *Design-to-Robotic-Operation* workshop at the *Hiperorgânicos 8 Symposium* [12] in Rio de Janeiro, Brazil. Accordingly, the authors acknowledge the supporting advice of Henriette Bier, Director of *Robotic Building, Delft University of Technology*; and the contributions of Marcela Sabino, Ricardo Weissenberg, and Eduardo Migueles, members of the *Museum of Tomorrow Laboratory*.

REFERENCES

[1] T. G. Stavropoulos, D. Vrakas, and I. Vlahavas, "A survey of service composition in ambient intelligence environments," *Artif Intell Rev*, vol. 40, no. 3, pp. 247–270, 2013.

[2] J. F. d. Paz Santana, V. Julian, G. Villarrubia, G. Marreiros, and P. Novais, *Ambient intelligence: Software and applications : 8th International Symposium on Ambient Intelligence (ISAmI 2017)*. Cham, Switzerland: Springer, 2017.

[3] P. R. Milgrom, "The economics of modern manufacturing: technology, strategy, and organization," *The American Economic Review*, vol. 80, no. 3, pp. 511–528, 1990.

[4] A. Liu Cheng and H. Bier, "Extension of a High-Resolution Intelligence Implementation via Design-to-Robotic-Production and -Operation strategies," in *Proceedings of the 35th International Symposium on Automation and Robotics in Construction (ISARC) 2018*, Berlin, Germany, 2018, pp. 1005–1012.

[5] A. Liu Cheng, "Machine Learning as enabler of Design-to-Robotic-Operation," *Archidoct*, vol. 6(1), no. 11, pp. 37–49, 2018.

[6] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "A Public Domain Dataset for Human Activity Recognition Using Smartphones," in *Proceedings of the 21th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN 2013.*, 2013.

[7] Google Cloud Platform®, *Cloud Speech-to-Text: Speech to text conversion powered by machine learning and available for short or long-form audio.* [Online] Available: https://cloud.google.com/speech-to-text/. Accessed on: Apr. 25 2018.

[8] European Association of Research and Technology Organisations (EARTO), *The TRL Scale as a Research & Innovation Policy TOOL: EARTO Recommendations.* [Online] Available: http://www.earto.eu/fileadmin/content/03_Publications/The_TRL_Scale_as_a_R_I_Policy_Tool_-_EARTO_Recommendations_-_Final.pdf. Accessed on: Jan. 07 2015.

[9] TensorFlow™, *An open source machine learning framework for everyone.* [Online] Available: https://www.tensorflow.org/. Accessed on: Apr. 20 2018.

[10] Google Cloud Platform®, *Cloud Vision API: Derive insight from images with our powerful Cloud Vision API.* [Online] Available: https://cloud.google.com/vision/. Accessed on: Apr. 20 2018.

[11] C. Follini, A. Liu Cheng, G. Latorre, and L. Freire Amores, "Design and Development of a Novel Robotic Gripper for Automated Scaffolding Assembly," in *Proceedings of the 3rd IEEE Ecuador Technical Chapters Meeting (ETCM) 2018*, 2018, in press.

[12] *Hiperorgânicos - Simpósio Internacional de Pesquisa em Arte, Hibridização, Biotelemática e Transculturalismo.* [Online] Available: http://www.nano.eba.ufrj.br/hiper8/. Accessed on: May 30 2018.