An Optical Guiding System for Gesture Based Interactions in Smart Environments

Martin Majewski[1], Tim Dutz[2], and Reiner Wichert[1]

[1] Fraunhofer Institute for Computer Graphics Research IGD, Darmstadt, Germany
{martin.majewski,reiner.wichert}@igd.fraunhofer.de
[2] Multimedia Communications Lab, Technische Universitaet Darmstadt, Germany
tim.dutz@kom.tu-darmstadt.de

Abstract. Using gestures to control Ambient Intelligence environments can result in mismatches between the user's intention and the perception of the gesture by the system. One way to cope with this problem is to provide the user with an instant feedback on what the system has perceived. In this work, we present an approach for providing visual feedback to users of Ambient Intelligence systems that rely on gestures to control individual devices within their environments. This paper extends our previous work on this topic [1] and introduces several enhancements to the system.

Keywords: Gesture-based Interaction, Visual Feedback, Ambient Intelligence.

1 Introduction

Since Mark Weiser formulated the vision of ubiquitous computing systems embedded pervasively in our everyday environments [2] back in 1991, the amount of intelligent networked-devices has grown significantly. They are present in the form of smart entertainment systems such as TVs and HiFi sets, embedded in home automation systems and white ware, or part of communication devices such as tablet computers and smartphones. Every single of these devices provides its own, specific user interface and this can make it difficult for the user to keep track of the wide variety of functionalities provided. Consequently, there is a growing interest for more comprehensive interaction methods [3]. In the past couple of years, scientists invented and examined different approaches to provide a more natural and unified way of interacting with smart environments [4], and a very convenient way of selecting and interacting with devices within smart environments are gestures [5].

Because gestures are often used in interactions between humans and usually correctly interpreted by a human counterpart, interacting with smart environments via gestures feels natural and intuitive. However, there can be a significant mismatch between the understanding of a gesture when performed by a person and the interpretation of the same gesture by a computer system. This mismatch results from a variety of reasons:

N. Streitz and P. Markopoulos (Eds.): DAPI 2014, LNCS 8530, pp. 154–163, 2014.

- An incorrect positioning of the gesture tracking sensors
- An insufficient tracking precision
- A wrong interpretation of the gathered tracking data by the computer system
- The user's misleading self-assessment when performing unambiguous gestures
- The user's erroneous believe in an unlimited adaptivity of the computer system

The creation of failsafe gesture recognition systems that are capable of covering large areas (such as the entire living room) is an enormous challenge as these systems have a high implementation complexity. An interim solution on the way towards this goal might be to develop systems that can provide users with instant, sophisticated feedback on what the system has perceived, thus enabling them to better adapt their behavior to the system's capabilities. To this end, we have developed an optical guiding device that acts like an omnipresent environmental cursor. This laser-based device visualizes the current interpretation of the user's gesture to her, thus allowing her to adapt accordingly. Furthermore, we implemented a highly customizable and flexible software solution that connects multiple economy-priced gesture and position tracking devices such as Microsoft's Kinect, the Leap Motion Tracking bar, and the CapFloor system for the provision of reliable multi-resolution user localization and gesture tracking.

2 Related Work

The research on whole-body gestures can be traced back to at least the early 1960s [6, 7]. The current efforts in this area concentrate mainly on virtual reality and entertainment applications [8, 9]. To perform gestural interaction, a human body pose recognition system is needed and since the release of Microsoft's Kinect sensor, 3D cameras that sell at a reasonable price have become widely available.

This text is an addition to our earlier work [1]. For this, we have found inspiration mainly in the research of Wilson et al [8], who first introduced the dedicated XWand input device based on inertial measurement units and infrared LEDs. These allow for the determination of the XWand's position and orientation in order to calculate the location that it is currently being pointed at by the user. Although not being a marker free whole-body interaction method in its own right, it led to Wilson's later work, the WorldCursor [1]. This laser-pointing device highlights the location currently selected by the XWand in the environment, thus improving the selection process.

The Beamatron project of Wilson et al. [2], published by Microsoft Research in 2012, shows a marker free interaction approach using several Kinect cameras, a microphone array setup, as well as a high definition projector mounted on a stage-light robot arm. Although being heavily related to our works with respect to the character of the utilized input and output devices, the Beamatron project is not a feasible solution for everyday home setups. The stage-light robot arm makes it a costly product, and it is too voluminous for the average ceiling height. Furthermore, although it relies on complex algorithms to identify and follow the user's location, it is relatively static and inflexible. Figure 1 shows pictures of both the WorldCursor (to the left) and the Beamatron (to the right).

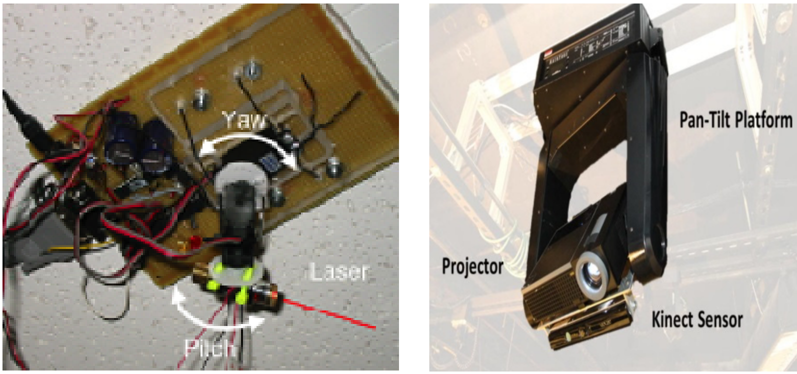


Fig. 1. WorldCursor, left [11] and Beamatron, right [12]

The effectiveness of locating the user in a real word environment highly depends on the sensor technology used. Camera based location is a common method nowadays. Since the introduction and the gaining popularity of smartphones and tablet computers, the focus of gesture recognition research oftentimes lies on capacitive sensory empowered gesture interaction. The transition from small screens to large areas capable of not only tracking smaller limbs, but the whole body was performed by Grosse-Puppendahl et al. [3] in 2013 with the OpenCapSense tookit and Braun et al. [4] with the CapFloor system – a highly affordable floor setup for locating people's position and approximating their posture within a given area. Another accurate solution for detailed limb tracking can be found with the LeapMotion IR sensor bar, although it is limited to a small spatial room.

With our solution, we show a compact, affordable, and relatively flexible visual feedback system. We additionally use the CapFloor technology for providing more accurate position estimation and combine this location approach with both a Kinect camera for low-resolution entire body gesture recognition and a LeapMotion device to support finger gestures.

3 The Perception Gap

The gap between a user's intention when performing a gesture and the system's interpretation of this gesture was already described at length in the predecessor work of Majewski et al. [1]. Due to this, we will only briefly summarize our findings here. In the subsequent paragraphs of this section, we will then introduce the modifications to the system as described in our previous paper. These modifications should help to close the perception gap to an even greater extent than the original system.

To use the index finger as a pointer and to thus generate an immaterial cone that spreads towards the pointing location is a common choice by humans when trying to point at something, but such a gesture can hardly be comprehended by gesture interpretation systems that only rely on simplified skeleton models, such as the Kinect camera. Based on such a simplified skeleton model, only the orientations of the larger limbs of the human body can be perceived the system and as such, when the user is pointing at something with her finger, the system actually bases its interpretation of the pointing location on the orientation of the user's shoulder and wrist. This often results in a significant mismatch between the intended pointing location and the gesture interpretation by the system. Figure 2 visualizes this problem.

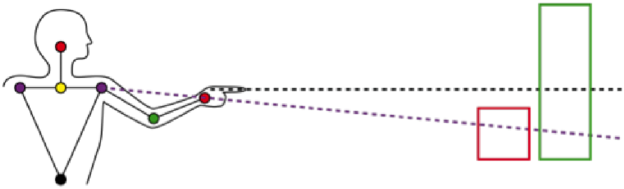


Fig. 2. Pointing mismatch

A second challenge for the computer-based interpretation of gestures is the parallax between gaze and arm angle that is affecting the perceived direction of a pointing gestures. Figure 3 shows this effect from a bird's perspective. There is a considerable difference between constructing the ray that spreads towards the pointing location from the shoulder and wrist on the one hand, and from the iris on the other. The closer the target object is, the larger the effect. In certain situations, this parallax is reduced, as highlighted on the right side of the figure. However, already a small offset angle results in a several centimeters large shift when pointing at something within a few meters distance. More specifically, an offset of merely five degrees between the user's gaze and her shoulder-wrist-line will result in a deviation of 17 cm when pointing at something in a distance of two meters. This error can make pointing on several relatively small devices in close proximity to each other very difficult.

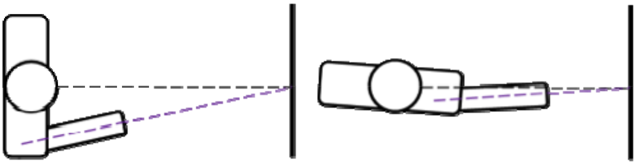


Fig. 3. Parallax mismatch

A third problem occurs when the time delay between performing a pointing action and the reaction by the system exceeds a certain time interval. According to a study by Kammer et al. [5], only delays of less than 100 milliseconds are perceived to be acceptable by users. As such, a user will be inclined to find a slow gesture interpretation system unsatisfactory, even if her gestures are interpreted correctly by the system and trigger the intended effects.

4 Visual Feedback System

4.1 Visual Feedback Robot

The visual feedback robot was introduced in 2012 by Majewski et al. [1] and is based on an Arduino microcontroller board, operating a small laser mounted on two servo-motors that allow for a free positioning of a laser dot inside a room. Figure 4 shows both an image of the robot alone (to the left) and it being mounted on the ceiling of a living room (to the right).

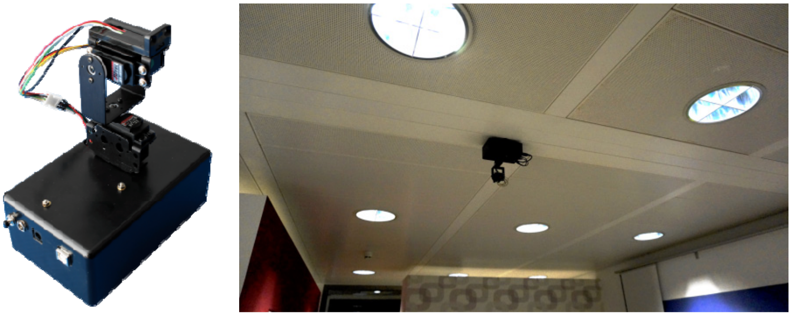
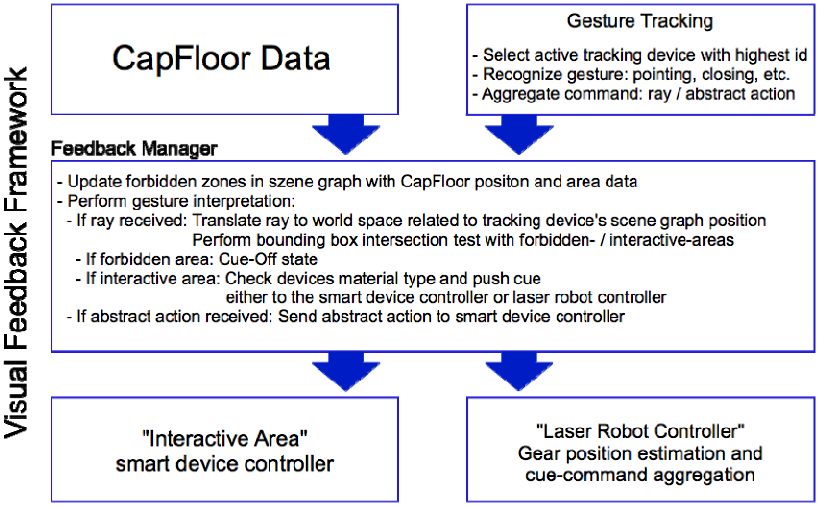


Fig. 4. Visual Feedback Robot (left) and mounted on living room ceiling (right)

4.2 Visual Feedback Framework

The software framework that supported the original feedback robot was completely redesigned, resulting in a modular architecture that allows extending functionality by providing binding modules for any kind of existing gesture input solutions. Through this, we were able to use both the LeapMotion sensor and the Kinect depth camera for gesture recognition, relying on the more detailed hand skeleton model provided by the LeapMotion for a more precise navigation in a smaller area by pointing a finger, while using the rougher whole-body skeleton of the Kinect to navigate long distances using arm gesture. This compensates the limitation of the Kinect camera not being able to track small limb joints. The tracking module of our approach abstracts the input devices and generates a unit ray representation. Every input device is associated with a priority ID to generate a hierarchical ordering of the provided tracking accuracy. Figure 5 provides an overview of the architecture of our framework. The various tracking methods used in our system are detailed in the next section.



- Select active tracking device with highest id
- Recognize gesture: pointing, closing, etc.
- Aggregate command: ray / abstract action

- Update forbidden zones in szene graph with CapFloor positon and area data
- Perform gesture interpretation:
  - If ray received: Translate ray to world space related to tracking device's scene graph position
    Perform bounding box intersection test with forbidden- / interactive-areas
    - If forbidden area: Cue-Off state
    - If interactive area: Check devices material type and push cue
      either to the smart device controller or laser robot controller
  - If abstract action received: Send abstract action to smart device controller

Fig. 5. Visual Feedback Framework architecture

5 Tracking and Localization Methods

5.1 Low-Resolution Whole-Body Gestures

While in the progress of orientation and navigation, the user tends to use less accurate gestures to bypass larger spatial areas and provoke an immediate response from the system. For these situations, we use a less detailed motion tracking approach based on the Kinect camera and create the resulting pointing ray from the shoulder-wrist-line. This is sufficiently accurate for many types of gestures, although it forces the user to adapt her gestures to the system's perception, suffering from the problems described in chapter 3. This is where the supporting effect of the instant feedback that is provided by the visual feedback robot as introduced in chapter 4 is the largest.

5.2 High-Resolution Hand-Based Gestures

Excessive limb motions, which are required for interacting with the low-resolution system, can be tedious or even impossible, when the spatial area available is highly limited. Furthermore, the ray construction by way of the shoulder-wrist-line is not as precise as in case of using the finger. For this reason, we have extended our gesture tracking system with a LeapMotion infrared sensor bar. Alas, the sensor bar has severe limitations in terms of detection range, which is an area of only about 0.226 m³ above the bar. Consequently, the amount of use cases that depend on this kind of sensor is strongly limited. We chose to use the LeapMotion as a stationary tracking device on the armrest of a sofa, were it is comfortable to use while sitting next to it. In this position, the lower arm is supported by the armrest, making the interaction with the bar comfortable. As soon as the user approaches the detection field of the LeapMotion, the cue's control instance is passed to the LeapMotion's skeleton results. Until the user leaves the detection field of the device, the pointing ray is constructed through the collateral ligament and the direction of the 3rd phalanx of the pointing finger. Figure 6 visualizes this principle.

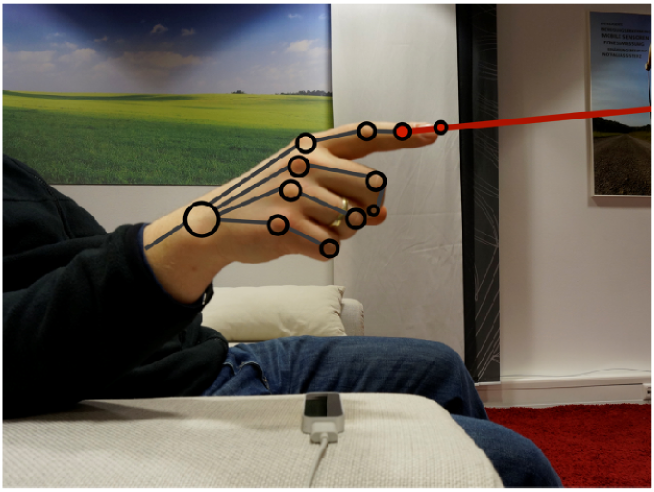


Fig. 6. High-resolution pointing gestures with LeapMotion

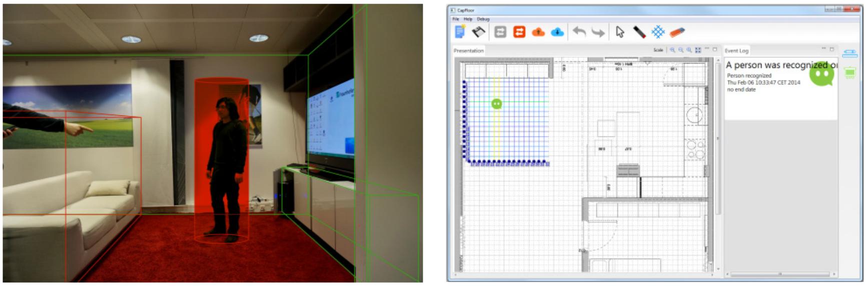
5.3 CapFloor User Position Detection

Camera-based tracking systems have their limitations in terms of field of view and tracking distance, and are also a cause for privacy concerns for many users. Furthermore, even powerful depth tracking camera systems have their limitations when it comes to the quantity of detectable users. To both provide for a better coverage of the user detection area and address these concerns, we have investigated a possibility to detect users through the CapFloor capacitive sensor based floor [13, 14].

The hardware demands of CapFloor suite our interest in delivering an economy-priced solution very well. The CapFloor uses thin copper wires as antennas, which are orthogonally arranged under a carpet or integrated into tile joints. These antennas are connected to a sensor bus system in the room's baseboard and the gathered data is processed within the CapSense framework. This framework, being capable of detecting and classifying standing or lying objects, provides us with the required data for our setup. We use this to realize two uses cases as described in the next two sections.

5.4 Dynamic Forbidden Area Determination

For obvious security reasons, we have designed our laser-based visual feedback system to avoid user eye contact. To this end, the system tracks the position of all users using and will not point the laser beam to any of such locations. Different to static objects like furniture, users tend to move through the room and as such, there position needs to be constantly updated. However, camera-based tracking approaches such as ones based on the Kinect are only of limited use for this, as the can move out of sight of their tracking area. To this end, we rely on the CapFloor sensor to reliable inform our system of the user's position. More specifically, we use the classification of the detected objects to make body size estimation. If the CapFloor software classifies the detected person as standing, we use a cylindrical bounding box and set its height to 200 cm. This height dimensions accords to the door's height in our environmental model. The width is set to the significantly detected floor proximity area's long side of the detected person, but not less than 50 cm in diameter to ensure the head coverage even if the person bends its head sideways. If the CapFloor software classifies the detected person as lying, we use a rectangular bounding box and set its height to 50 cm. The width and depth dimensions are determined by the side length sum of the active antenna cells. These bounding boxes are updated in real-time in our environmental model and ensure the dynamic creation of forbidden areas not to be highlighted by the visual feedback robot.



A person was recognized o
Person recognized
Thu Feb 06 10:33:47 CET 2014
no end date

Fig. 7. Forbidden area determination

5.5 Selective Device Activation

If a system is supposed to cover a larger with visual pointing feedback, it requires multiple cameras and projectors. Leaving all those devices active and just waiting for the user to enter the camera's field of view is certainly not a satisfying situation. Based on the user location detection with CapFloor, our approach allows the activation of cameras and projectors only when the user is close enough to benefit from the functionalities that they can provide.

6 Conclusion and Future Work

In this contribution we have introduced three important additions to our visual feedback system that compensate the limitations of the low detail skeleton reconstruction of the Kinect camera, made possible through the development of a highly flexible visual feedback framework. Furthermore, we have increased the security and usability aspect of the system by providing static and dynamic forbidden areas that avoid unwanted cue projection.

To measure the benefits of our current work, we intend to perform an extensive user evaluation of our system in the near future. The users' feedback will then be used to improve the system further. We also plan to investigate a richer cue provision with portable multimedia projectors, as well as more complex laser setups where we focus on affordability and suitability for daily use.

Acknowledgments. This work is partially financed by the European Commission under the FP7-ICT-Project Miraculous Life (grant agreement no. 611421).

References

1. Majewski, M., Braun, A., Marinc, A., Kuijper, A.: Visual support system for selecting reactive elements in intelligent environments. In: Cyberworlds (CW). IEEE, Darmstadt (2012)
2. Weiser, M.: The Computer for the 21st Century. Scientific American 265(3) (1991)
3. Sears, A., Jacko, J.A. (eds.): The human-computer interaction handbook: Fundamentals, evolving technologies and emerging applications. CRC Press (2007)
4. Reeves, L.M., Lai, J., Larson, J.A., Oviatt, S., Balaji, T.S., Buisine, S., Collings, P., Cohen, P., Kraal, B., Martin, J.-C., McTear, M., Raman, T.V., Stanney, K.M., Su, H., Wang, Q.Y.: Guidelines for multimodal user interface design. Communications of the ACM 47 (2004)
5. Grguric, A., Mosmondor, M., Kusek, M., Stockloew, C., Salvi, D.: Introducing gesture interaction in the Ambient Assisted Living platform universaal. In: ConTEL 2013: Proceedings of the 12th International Conference on Telecommunications, Zagreb (2013)
6. Heilig, M.L.: Sensorama Simulator. US Patent 3050870, US Patent Office, Long Beach (1962)
7. Sutherland, I.E.: Sketchpad: A man-machine graphical communication system. In: Afips Conference Proceedings, vol. 2(574). ACM, New York (1963)
8. Kessler, G.D., Hodges, L.F., Walker, N.: Evaluation of the CyberGlove as a whole-hand input device. ACM Transactions on Computer-Human Interaction 2(4) (1995)
9. Gallo, L., De Pietro, G., Marra, I.: 3D interaction with volumetric medical data: Experiencing the Wiimote. In: Proceeding Ambi-Sys 2008, Proceedings of the 1st International Conference on Ambient Media and Systems, vol. (14). ICST, Brussels (2008)
10. Wilson, A., Shafer, S.: XWand: UI for intelligent spaces. In: Proceedings of the ACM Conference on Human Factors in Computing Systems, vol. (5). ACM, NewYork (2003)
11. Wilson, A., Pham, H.: Pointing in intelligent environments with the worldcursor. In: INTERACT International Conference on HumanComputer Interaction. IOS Press, Ohmsha (2003)
12. Wilson, A., Benko, H., Izadi, S., Hilliges, O.: Steerable Augmented Reality with the Beamatron. In: Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology. ACM, NewYork (2012)
13. Große-Puppendahl, T.A., Berghoefer, Y., Braun, A., Wimmer, R., Kuijper, A.: OpenCapSense: A rapid prototyping toolkit for pervasive interaction using capacitive sensing. In: PerCom 2013, San Diego (2013)

14. Braun, A., Heggen, H., Wichert, R.: CapFloor - A Flexible Capacitive Indoor Localization System. In: Chessa, S., Knauth, S. (eds.) EvAAL 2011. CCIS, vol. 309, pp. 26–35. Springer, Heidelberg (2012)
15. Kammer, D., Keck, M., Freitag, G., Wacker, M.: Taxonomy and Overview of Multi-touch Frameworks: Architecture, Scope and Features. In: Workshop on Engineering Patterns for Multitouch Interfaces, Berlin (2010)