

University of Groningen

## Presenting using Two-Handed Interaction in Open Space

Vlaming, Luc; Smit, Jasper; Isenberg, Tobias

*Published in:*

THIRD ANNUAL IEEE INTERNATIONAL WORKSHOP ON HORIZONTAL INTERACTIVE HUMAN-COMPUTER SYSTEMS, PROCEEDINGS

**IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.**

*Document Version*

Publisher's PDF, also known as Version of record

*Publication date:*

2008

[Link to publication in University of Groningen/UMCG research database](#)

*Citation for published version (APA):*

Vlaming, L., Smit, J., & Isenberg, T. (2008). Presenting using Two-Handed Interaction in Open Space. In THIRD ANNUAL IEEE INTERNATIONAL WORKSHOP ON HORIZONTAL INTERACTIVE HUMAN-COMPUTER SYSTEMS, PROCEEDINGS (pp. 31-34). NEW YORK: IEEE (The Institute of Electrical and Electronics Engineers).

### Copyright

Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

### Take-down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): <http://www.rug.nl/research/portal>. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.

# Presenting using Two-Handed Interaction in Open Space

Luc Vlaming      Jasper Smit      Tobias Isenberg  
University of Groningen, The Netherlands  
{asperientje|jbsmit}@gmail.com, isenberg@cs.rug.nl

## Abstract

*Based on recent demonstrations of low-cost, infrared-based point tracking, we explore two-handed, surface-less interaction for presentation. On both hands, thumb and index finger are equipped with retro-reflective markers which are tracked by a Wiimote. We contribute a robust finger pairing and pinch recognition method that allows us to discriminate the hands and to initiate actions. We apply this input to a presentation application that allows users to work with slide decks, images, and videos. We identify specific requirements of this application domain and discuss the implemented transformation interactions and widgets. We report on user experience in both casual use and an actual presentation as well as discuss advantages and limitations.*

## 1. Introduction

Multi-touch interaction with small and large displays has received much attention in recent years. The tracking of multiple points typically requires elaborate hardware setups such as Dynawall [5], DiamondTouch [4], FTIR [6], DViT, or the technique used in Microsoft's Surface technology. Recently, Johnny Lee demonstrated how to use Nintendo's Wii controller (Wiimote) to simultaneously track up to four points that are emitting or reflecting infrared (IR) light [9]. In particular, he showed how to build a low-cost setup to track fingers in open space for interacting with a vertical 2D environment, similar to a virtual wall display.

We extend this concept by exploring how to reliably distinguish between two hands, tracking the thumb and index finger per hand for both motion and grabbing gestures, inspired by the fictional interaction shown in the *Minority Report* movie. While there are some related projects such as Cynergy Labs' *Project Maestro*<sup>1</sup> or the Fraunhofer *iPoint Presenter*,<sup>2</sup> we focus on using low-cost passive reflectors that are attached to the fingers. We describe how to reliably track the finger pairs with this setup and discuss how to detect pinching actions for selection. We apply our technique to interacting with images, image stacks, and videos within a presentation system that uses these elements. We

<sup>1</sup>Proprietary development; see <http://labs.cynergysystems.com/>.

<sup>2</sup>See <http://www.hhi.fraunhofer.de/index.php?1401&L=1>.

describe the necessary widgets and integration of interaction techniques such as RNT [8] and two-point manipulation [7]. We report on the use of our system for a 30 minute presentation and on the comments from five novice users.

Our contributions, thus, are an algorithm for robust finger pairing and pinching recognition for *two-handed four-finger interaction* with a discussion of its performance, the demonstration of this technique in a presentation application including the analysis of its constraints and advantages within this application domain, and an informal evaluation of the system. This technique allows users to indirectly interact with objects on large presentation screens from a distance, possibly with the help of a separate small display.

## 2. Related Work

The systems that are most related to our techniques are the previously mentioned *Project Maestro* and *iPoint Presenter*. The former uses a similar setup with a Wiimote but relies on special gloves with extra hardware attached, whereas we restrict ourselves to using retro-reflective fingertips and, hence, require only a source of infrared light. *iPoint Presenter* can even do without either active or passive markers but instead uses a special hardware setup to track a person's hands. Other techniques allow people to interact with digital walls or augmented desktops using their bare hands [11, 12] or control presentations using a dataglove-tracked hand [1]. In all cases, people can manipulate 2D objects using a set of gestures, in the case of *Project Maestro* pinching with thumb and index finger. Such pinching interaction has previously mostly been used in VR applications [3], where special gloves with sensors on multiple fingers make complex selection or menu interactions possible.

In pinching selection in our system relates to the *dual finger midpoint* technique for precise selection [2]. However, we use pinching interaction only for grabbing objects and initiating actions. In addition, we use previously presented geometric transformation methods for rotation of objects [7], but also for translation and scaling. Related to controlling video as one of the specific types of media we work with, are fluid interaction techniques such as presented by Ramos and Balakrishnan [10]. While we concentrated mainly on implementing general interaction techniques (se-

lection and transformation), such methods and more general ones could be implemented within our framework.

### 3. Robust Point Pairing and Pinch Detection

Our two-handed interaction uses the space directly in front of a person that is within reach. Unlike in 3D tracking, we are only interested in the 2D location of hands and fingers, much like in the 2D interaction on large vertical displays. Using the setup suggested by Lee [9], we track the positions of thumb and index finger of both hands—marked with retro-reflective material attached to the finger tips of gloves (Fig. 1). Like Lee, we use an IR light source and a Wiimote as a low-cost, high-frequency IR camera setup.

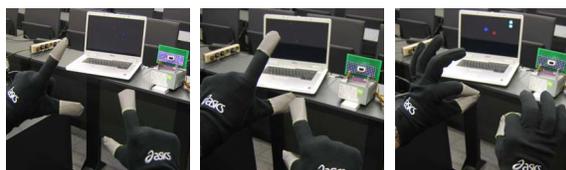
Lee used a simple show or no-show technique to activate and deactivate interaction points [9], somewhat like many input techniques on tabletop and wall displays. While effective in such settings due to the direct visual integration of the interacting finger/hand or pen with the display, in our setting the interaction occurs indirectly—users have a separate screen such as a laptop at a certain distance in front of them (see Fig. 2). Thus, it is necessary to show the tracked points on the screen for feedback and to have a defined activity to select objects or initiate actions. For these we were inspired by the pinching interaction used in VR [3].

To implement hand and gesture recognition we employ the *wiiuse* library.<sup>3</sup> It provides four points detected by the Wiimote, some possibly marked as invisible. The assignment of points can change, e. g., when a point changes its visibility or when points are spatially too close to each other causing them to merge into one. Thus, the location of tracked points is known without a reliable identification. Therefore, to be able to identify separate hands, the tracked points need to be identified and paired robustly; then, pinching interactions need to be detected to select items.

For an initial and all subsequent pairing we use an incremental approach. The first two points that are detected are each assigned to a different pair. When adding the next point, a reassignment may occur. For this, each possible pairing is computed and the one with the closest distance is chosen. We calculate a weighted Euclidean distance with the vertical axis favored over the horizontal one (the  $x$ -distance is multiplied by a factor of 1.8). This is based on the observation that, when people hold their arms in front of them with the fingers stretched out, the tips of thumb and index finger almost form a vertical line (Fig. 1(a)). The fourth point is finally added to the remaining incomplete pair.

The advantage of this incremental method over a naïve closest points pairing becomes evident when points have to be removed, which may be caused by a number of events. Points may actually be missing in the image (Case 1), e. g., due to occlusion of one finger tip or due to one finger be-

<sup>3</sup>See <http://www.wiiuse.net/>.



(a) Hands stretched out for interacting. (b) Both hands close so points merge. (c) Hands pinching, initiating actions.

Figure 1. Pairing and pinching.

ing outside of the camera’s field of view. Alternatively, two points from different pairs may come too close to each other (Case 2; Fig. 1(b)) so that only one large “blob” and, hence, only one point is detected. The same effect also occurs when the two fingers of one hand are moved together to initiate an action by “pinching” (Case 3; Fig. 1(c)). While all these effects look similar in the input, they need to invoke different actions. In addition, when points reappear they again have to be associated to the correct pairs.

We deal with Case 1 by removing points that have disappeared and adding them again when they reappear. If multiple points disappear and then reappear, the incremental algorithm takes care of pairing. The problem of correct pairing is most evident in Case 2, when two fingers from different pairs come close to each other. For the Wiimote, they then form a single point and when the two fingers come apart again a wrong pairing may happen. While this occurs rarely in practice,<sup>4</sup> we deal with the remaining cases by looking at the motion of the points in the pairs after a fourth point has been added. If the points in both pairs move in roughly opposite directions ( $> 120^\circ$ ) in the following 0.3 s, the two points with the most similar directions are paired and the other two are put in the second pair. Finally, pinching is distinguished from disappearing points by thresholding the distance of each pair’s two points. If the points were closer together than the threshold before one is removed, we mark this as a pinching event for initiating actions. Otherwise, we proceed as before. This procedure leads to a robust pairing of the four points and a stable distinction of pinching from other lost points events. Thus, the processed input can now be used for *two-handed four-finger interaction*.

### 4. Case Study—Presentation Application

To demonstrate the applicability of this interaction technique, we designed a setting in which a presenter can show-case documents, movies, or images—inspired by the *Minority Report* movie. The physical setup (Fig. 2), similar to a regular presentation setup, has the presenter facing the audience and a screen on which he or she sees the interface. The Wiimote and the IR light source are positioned close to the

<sup>4</sup>We believe the Wiimote performs internal processing on the IR ‘blobs’ with leads to sub-blob accuracy, thus yielding more stable point IDs.



Figure 2. Presentation setup.

screen, facing and illuminating the presenter. The presentation is also projected onto the wall behind the presenter.

#### 4.1. Interactive Transformations

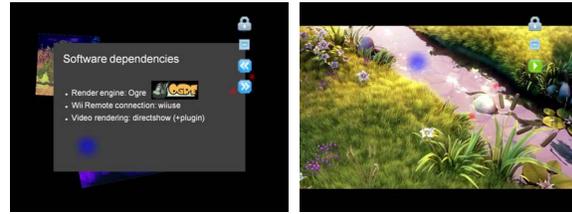
Our two-handed input technique allows users to interact using either only one hand or both hands. For realizing translation, rotation, and resizing of objects using the two interaction metaphors, we implemented both RNT [8] as well as two-point rotation [7]. Both techniques integrate rotation and translation transformations, while the two-point rotation also permits scaling. Finally, switching between objects occurs when they are grabbed by at least one point, bringing them in front of all other objects.

For the presentation application we restricted the resizing of objects in order to avoid that objects get too small to be enlarged again, or too large to require too many scaling steps to shrink them below screen size again. In addition, accidental grabbing actions may occur which can invoke unwanted transformations that could disrupt a presentation. To avoid these, each object can interactively be locked from transformations, only permitting bring-to-front actions.

#### 4.2. Elements and Widgets

As noted above, we focused on the fundamental elements that enable presentations: slide/image decks, individual images, and videos. For all elements we provide a number of common actions represented through widgets (see Fig. 3). These widgets are only shown when an object is active, i. e., grabbed by at least one hand. The other hand can then be used to grab one of these widgets to initiate its actions.

The common actions comprise the locking/unlocking mechanism, minimizing the object to a location at the bottom of the screen, and restoring them to their previous positions. In addition, specific object types have additional specialized functions including slide flipping for slide decks,



(a) Widgets to control slides. (b) Widgets to control videos.

Figure 3. Elements and widgets for presenting.

and starting/pausing a video as well as rewinding it (Fig. 3).

For videos we also control their sound level because many videos playing at the same time would otherwise be irritating. Non-selected videos are reduced in volume to  $-17$  dB. We decided to not completely mute them to give users additional awareness that other videos are playing.

The control widgets are shown always at the same position on the interface's perimeter (Fig. 3). Compared to showing the widgets on a location relative to each object, this allows presenters to more easily remember the locations of the controls. In addition, this position also allows us to make the widgets larger and, thus, easier to select without interfering much with the elements on the interface.

#### 4.3. Informal Evaluation and User Experience

In our experience, the two-handed interaction with the implemented presentation system works well. The Wiimote recognizes the interaction points reliably in distances from 0.5 m to about 3 m for our setup of 111 IR-LEDs for illumination and well-pronounced, well-reflecting markers on the fingers. The markers can be quite large, ours cover the top 46 mm of each finger. The threshold for pinching detection needs to be adapted to the average distance from the camera since it is of angular nature and the camera cannot detect the distance of the points to itself. We did not notice gaps (other than actual occlusions) or lag in the tracking data and the tracked positions result in smooth point motions on the screen without any additional processing.

We evaluated the usage of the system in an actual presentation of approx. 30 minutes. By adjusting the interaction space so that the presenter could hold the upper arms parallel to the body and the lower arms in a  $90^\circ$  angle, he did not get too tired. This may be because the arms do not need to be held up for the entire time, only for actually interacting with the slides. In terms of the interaction, we found that the system enabled pointing to objects on the screen using the tracked hands more easily than with, e. g., a laser pointer. Our input detection was robust and allowed for a smooth presentation. The presenter could freely structure the presentation and did not to follow a predefined slide deck.

In addition, we asked five novice users (2 female, 3 male) to try out our system for about 15–20 minutes each. People

with previous computer and presenting experience needed only about five minutes to become familiar with the technique. They generally enjoyed the new interaction technique, with some people calling it “awesome” or “cool, definitely cool.” Other people with little to no previous computer experience required more time to get used to the technique. They reported that it “first feels strange,” probably due to the required mental mapping from fingers to points on the screen that people with more computer experience are used to as they interact via mice more frequently. This also seems to be supported by the comment of one person that the interaction technique “doesn’t have much of a learning curve, as much a learning curve as a mouse.”

We did, however, notice that some people initially had problems with understanding the two-dimensional character of the interaction, trying to initiate actions by moving their hands in the depth direction. Some people also had difficulties with grasping the concept that both hands were necessary to initiate some actions including both the two-point rotation and the actions associated to widgets.

We also realized that a careful setup of the system is important because otherwise it becomes necessary to use the hands in uncomfortable positions, in particular, if one tries to reach locations at the perimeter. This problem could be alleviated by reducing the part of the visible field of view of the camera that is actually used for control. One could start the interaction with a brief calibration step in which the four corners of the active interaction rectangle would be defined.

## 5. Summary and Conclusion

We examined two-handed four-finger interaction in open space using low-cost hardware. This supports indirect, distant interaction with presentation screens. In particular, we developed heuristics to robustly track the thumb and index finger of both hands as associated pairs and to recognize pinching actions of each pair. We applied this interaction technique to presentations in which the presenter can not only advance through slides but also interact with all objects on the screen, including translations, rotations, and scaling. For this purpose we implemented RNT and two-point rotation as well as a number of widgets to control the objects on the screen, some of them specific to the object class. Feedback from informal demonstrations of the systems suggest that people enjoyed the techniques and system but also pointed out a few shortcomings.

There are also a number of other limitations. One characteristic of the technique is that the interaction points need to be visible in order to give feedback to the users, similar to the mouse pointer on desktop operating systems, which maybe disturbing at times. Also, some training is necessary in order to avoid repeated pinching actions while the presenter is speaking, which may also be caused by the arms being lowered for rest so that, from the view of the camera,

the thumb and index finger come too close to each other. This points to another problem which is that arms do get tired when using the system over extended periods. We have shown, however, that presentations of about 30 minutes are possible. One interesting effect we observed is that the system forces people to constantly face the feedback screen and, thus, the audience, much more than typical two-screen setups and wireless presentation tools do. This may be used for presentation training for people who have the tendency to turn around for looking at the projection behind them.

In the future, we would like to look into using regular webcams with infrared filters instead of the Wiimote. Besides having a more stable wired connection, multiple webcams would allow us to track the points in 3D space, making more complex interaction techniques possible. In addition, it would be interesting to examine how the developed techniques would function in multi-display scenarios consisting of both horizontal and vertical surfaces.

## References

- [1] T. Baudel and M. Beaudouin-Lafon. CHARADE: Remote Control of Objects using Free-Hand Gestures. *Communications of the ACM*, 36(7):28–35, July 1993.
- [2] H. Benko, A. D. Wilson, and P. Baudisch. Precise Selection Techniques for Multi-Touch Screens. In *Proc. CHI*, pp. 1263–1272, New York, 2006. ACM Press.
- [3] D. A. Bowman, C. A. Wingrave, J. M. Campbell, V. Q. Ly, and C. J. Rhoton. Novel Uses of Pinch Gloves™ for Virtual Environment Interaction Techniques. *Virtual Reality*, 6(3):122–129, Oct. 2002.
- [4] P. Dietz and D. Leigh. DiamondTouch: A Multi-User Touch Technology. In *Proc. UIST*, pp. 219–226. ACM Press, 2001.
- [5] J. Geißler. Shuffle, Throw or Take it! Working Efficiently with an Interactive Wall. In *Proc. CHI*, pp. 265–266, New York, 1998. ACM Press.
- [6] J. Y. Han. Low-Cost Multi-Touch Sensing through Frustrated Total Internal Reflection. In *Proc. UIST*, pp. 115–118, New York, 2005. ACM Press.
- [7] M. Hancock, F. D. Vernier, D. Wigdor, S. Carpendale, and C. Shen. Rotation and Translation Mechanisms for Tabletop Interaction. In *Proc. TableTop*, pp. 79–86, Los Alamitos, 2006. IEEE Computer Society.
- [8] R. Kruger, S. Carpendale, S. D. Scott, and A. Tang. Fluid Integration of Rotation and Translation. In *Proc. CHI*, pp. 601–610, New York, 2005. ACM Press.
- [9] J. C. Lee. Tracking Your Fingers with the Wiimote, 2008. Web page <http://www.cs.cmu.edu/~johnny/projects/wii/> and online video, visited June 2008.
- [10] G. Ramos and R. Balakrishnan. Fluid Interaction Techniques for the Control and Annotation of Digital Video. In *Proc. UIST*, pp. 105–114, New York, 2003. ACM Press.
- [11] M. Ringel, H. Berg, Y. Jin, and T. Winograd. Barehands: Implement-Free Interaction with a Wall-Mounted Display. In *CHI Short Papers*, pp. 367–368. ACM Press, 2001.
- [12] C. von Hardenberg and F. Bérard. Bare-Hand Human-Computer Interaction. In *CHI Posters*. ACM Press, 2001.