Designing a multi-purpose capacitive proximity sensing input device

Andreas Braun
Fraunhofer IGD
Fraunhoferstr. 5, Darmstadt, Germany
andreas.braun@igid.fraunhofer.de
+49 6151 155 208

Pascal Hamisu
Fraunhofer IGD
Fraunhoferstr. 5, Darmstadt, Germany
pascal.hamisu@igid.fraunhofer.de
+49 6151 155 579

ABSTRACT
The recent success of Nintendo’s Wii and multi-touch input devices like the Apple iPhone clearly shows that people are more willing to accept new input device-technologies based on intuitive forms of interaction. Gesture-based input is thus becoming important and even relevant in specific application scenarios. A sensor type especially suited for natural gesture recognition is the capacitive proximity sensor that allows the detection of objects without any physical contact. In this paper we extend the input device taxonomy by Card et al to include this detector category and allow modeling of devices based on advanced sensor units that involve data processing. We have created a prototype based on this modeling and evaluated its use regarding several application scenarios, where such a device might be useful. The focus of this evaluation was to determine the suitability of the device for different interaction paradigms.

Categories and Subject Descriptors
B.4.2 [Input/Output and Data Communications]: Input/Output Devices - Data terminals and printers.
H.5.2 [Information Interfaces and Presentation]: User Interfaces - Evaluation/methodology, Graphical user interfaces, Input devices and strategies, Interaction styles.

General Terms
Design, Verification.

Keywords
Input devices, interaction design, modeling, evaluation.

1. INTRODUCTION
Natural interaction has been a strong research focus in human computer interaction the last few years. The capacitive proximity sensor is a detector able to sense the presence of objects without physical contact over a distance ranging from a few centimeters up to several meters, depending on sensing system and setup. This characteristic is allowing this detector type to recognize spatial gestures performed by a user.

There have been various attempts on creating non-haptic human-computer interfaces based on capacitive proximity sensors. However they are mostly limited to a few supported hand gestures [16] or only support applications [15].

In previous work we have followed a similar approach, using a single sensor [2] in order to control simple demonstration applications. Our focus for this paper was to go a step further and create a generic input device based on an array of capacitive proximity sensors that should be able to track the location of the user’s hand. The idea was to create a multi-purpose device that can be used in various application scenarios.

Since we wanted to go beyond previous technology-driven attempts we were looking for a way to formally model such devices and integrate them into existing input device taxonomies. Therefore we have evaluated several classical approaches focusing on their ability to allow the modeling of novel input devices. Based on this evaluation we decided to extend the modeling system and taxonomy of Card et al [5]. This includes the specific addition of capacitive proximity sensors but also a general methodology to simply modeling of modern input devices that rely on data pre-processing, which is difficult or even impossible to describe with the existing method.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Copyright 2011 ACM 978-1-60558-246-7/08/04...$5.00

Figure 1. Conceptual rendering of hand interacting with the active area of hand-tracking input device

Using these additions we have formally modeled a hand-tracking prototype based on an array of capacitive proximity sensors. Various pre-processing algorithms generating high-level information from raw sensor data have been tested on this prototype, in order to evaluate the functionality of such a device in several usage scenarios.

Unlike mechanical input devices the user receives no haptic feedback from this device; there is no feedback when entering the interaction area above the device. In addition there is also no direct visual feedback from the interaction area, where the user is directly touching the items he wants to manipulate, as in touch-
screens or tangible interfaces. Despite those challenges we have determined various application scenarios that can profit from such an input device:

- Unobtrusive media center control
- Medical application in sterile environments
- Interaction device for the physically challenged

In the final part of the paper we evaluate each of those scenarios using different demonstration applications with distinct interaction paradigms and interfaces:

- Gesture-controlled image viewer
- Menu-based, movement-controlled 3D object viewer
- Position-based regional image & document zoom

We have done the initial evaluation by gathering qualitative feedback from expert users that are experienced in the usage of various input devices

2. RELATED WORK

2.1. Natural Interaction

Diverse research projects of the last decade have investigated the next evolutionary steps in HCI, trying to achieve a more natural interaction between user and machine, i.e. through gestures, speech and physical interaction [13]. Popular approaches include microphone arrays to recognize voice and locate the user [12], multiple video cameras using feature extraction to identify user movements [15] and large active display areas capable of identifying user touches [10]. The of gesture control schemes in gaming is a very notable achievement. Nintendo is market-leader with the Wii console. Microsoft (Kinect) and Sony (Move) are bringing similar systems to market in fall 2010. Another very successful gestural input device is the Apple iPhone that has been influencing user interface and interaction design in mobile multimedia devices ever since its introduction in 2007.

2.2. Input Taxonomies

The growing number of input devices has led to attempts in categorization of those, creating various taxonomies of input. An early user-centric system by Foley et al [7] is based on the user’s intentions that have been grouped into six categories of generic transactions. Buxton et al [3] used an approach restricted to hand operated devices that tries to categorize devices based on features of the human sensory system and provided degrees of freedom. A final classical approach is modeling the physical properties of each sensor attached to the input device and combines them using several composition operators. This system has been developed by Card et al [5].

2.3. Capacitive Proximity Sensors

Capacitive proximity sensing is a rather old application that first appeared in the 1920s [8]. Throughout the 1990s it has been a research interest at the MIT Media Lab [11]. Research of those systems is ongoing, with several commercial and scientific systems available [6], [4].

3. INPUT DEVICE MODELING

While being an established sensing device, it has not yet been tried to integrate input devices based on capacitive proximity sensors into common input device modeling systems and taxonomies. We have evaluated the three systems mentioned earlier regarding the following criteria:

- Flexibility: Adjustability towards layout changes in devices (e.g. traditional mouse vs. scroll wheel mice)
- Generality: The scope of included input devices. The majority of input devices should be included.
- Extensibility: Simplicity of adapting the taxonomy regarding new device categories.
- Definiteness: Ambiguous device categorization should be avoided.

Table 1. Evaluation of input device taxonomies

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Foley</th>
<th>Buxton</th>
<th>Card</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flexibility</td>
<td>No combined devices</td>
<td>Only continuous hand-operated devices</td>
<td>Composition Virtual input devices</td>
</tr>
<tr>
<td>Generality</td>
<td>No audio, vision</td>
<td>No foot, audio, vision</td>
<td>No audio, vision</td>
</tr>
<tr>
<td>Extensibility</td>
<td>If in six defined categories</td>
<td>If continuous hand-operated device</td>
<td>If input, output domains can be defined</td>
</tr>
<tr>
<td>Definiteness</td>
<td>Multiple categorization for single devices</td>
<td>Supported devices well-defined</td>
<td>Device-based modeling</td>
</tr>
</tbody>
</table>

The taxonomy of Foley et al was discarded early, as the strict categorization makes it difficult to integrate devices based on novel interaction forms, like gestural input. Another interfering factor is the ambiguous categorization of multi-purpose devices. Buxton et al allows for good categorization but is not well-suited for versatile sensors like capacitive proximity sensors that can track a hand, but also feet or the whole body. Also multi-purpose devices that may be configured for various degrees-of-freedom would lead to ambiguous categorization. In conclusion the taxonomy and modeling by Card et al was chosen as candidate to integrate the new sensors and input devices. It allows for modeling various systems based on the same sensor platform by using a different composition. However the main advantage for our purposes is the virtual input device that can be modified to provide an elegant way to model complex data pre-processing, which will be shown later in this section.

In order to keep this document self-contained, the modeling system of Card et al is described briefly using an example. The approach is to model all input devices from singular input devices, usually sensors determining a certain physical property. These singular devices are described using a six-tuple \((M, In, s, R, Out, W)\), consisting of a manipulation operator \(M\), the input domain \(In\), system state \(s\), a resolution function \(R\), transforming the input domain into output domain \(Out\) and a final attribute \(W\) that acts as placeholder for additional functionality. The x-Movement wheel in a mechanical mouse has the six-tuple \((dPs, N_{real}, s, f_{lin}, N_{real}, −)\). A change in x-direction is measured as a real number, transferred using a linear function, leading to a real number output and no additional information.
Figure 2. Compositional features of the Card et al taxonomy

Figure 2 displays the compositional features of the taxonomy by showing the example of a mechanical mouse. The inputs of x-direction and y-direction sensors are merged into a xy-detector. Several other sensors, like buttons and scroll-wheel are added to the same device, leading to the multi-function devices that are very common today. Finally the device is connected to a virtual input device, the mouse cursor, prevalent in many operating systems to control graphical user interfaces.

Figure 3. Principle of capacitive proximity sensing

3.1. Capacitive proximity sensors

The principle of a capacitive proximity sensor is to measure the change in capacitance of an electrode, caused by objects passing through the generated field. A simplified version of the principle is displayed in Figure 3. An electrode is periodically charged and discharged. If a conductive object passes through the electric field the capacitance of the system increases. By measuring the charge flowing back from the electrode the relative change in capacitance can be detected. Assuming the conductive object is of a uniform size, this change is roughly proportional to the distance between electrode and object, following the law of capacitance between parallel plates.

\[ C = \varepsilon_0 \varepsilon_r \frac{A}{d} \]

The actual electrical properties of the sensing system are depending on electrode geometry, the detected object geometry, as well as various environmental factors like air humidity or disturbance from nearby electrical circuits. Solving such a system using electrostatics is highly complex and not practicable [1].

Table 2. Six-tuples according to Card taxonomy for both proximity sensor modes

<table>
<thead>
<tr>
<th>Mode</th>
<th>M</th>
<th>In</th>
<th>s</th>
<th>R</th>
<th>Out</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proximity Mode</td>
<td>( P_r )</td>
<td>Real Number</td>
<td>S</td>
<td>Transfer Function</td>
<td>Real Number</td>
<td>-</td>
</tr>
<tr>
<td>State Mode</td>
<td>( P_r )</td>
<td>((0, \ldots, n))</td>
<td>S</td>
<td>Static Function</td>
<td>((0, \ldots, n))</td>
<td>-</td>
</tr>
</tbody>
</table>

The manipulation operator \( M \) is a position \( P_r \) in spherical coordinates, relative to the position of the electrode. A single sensor is not able to detect the direction of the approaching object, thus determining angles requires input from multiple singular sensors.

3.2. Modeling complex data processing

Many modern input devices include microprocessors that allow for more or less complex data pre-processing before any events are sent to the actual computer. One example of such a system is the optical mouse that uses optical flow algorithms to detect movement and is send calculated xy-movement to the attached computer [9].

Figure 5. Data processing virtual input device connected to physical input device output and computer

Our idea was to integrate complex data processing into the Card taxonomy by encapsulating it into a virtual input device, placed between physical input device and computer (Figure 5). It
transforms data generated by all singular or merged input devices into new output, exceeding the scope of the resolution function described in the original version. Incoming data is put into a data structure we call Fusion Structure and transformed using various algorithms resulting in an output data structure. We suggest the notation in Figure 6 for a more detailed description of the data processing virtual input device.

Figure 6. Data processing virtual input device detailed

The input device is illustrated as grey box that shows applied algorithms and some additional information, without describing the process in full detail.

3.3. Proximity box modeling

Designing our input device we tried to meet the following requirements:

- Suitability for various application scenarios
- Easy accessibility for setup changes
- Balance between cost technical sophistication

Therefore we decided on a portable device housing six sensor units. Tests have shown that this number of detectors is providing a good balance between cost and functionality.

Thus we model the device from six capacitive proximity sensing singular input devices and a single data processing virtual input device. Figure 7 shows the composition model according to the extended Card model. The six capacitive proximity sensors are merged to a single sensor array that is sharing a layout. Each sensor is assigned a specific position in the array. The combined output is connected to the data processing virtual input device.

Figure 7. Proximity box composition according to extend Card model

As described earlier the data processing virtual input device is transforming the input located in the fusion structure into an output structure that is sent to the next device. The fusion structure is composed of six sensor data structures, containing the sensor value, a unique sensor ID and sensor location in xy-coordinates.

The additional positioning information is required to determine the hand position. The calculation uses different approaches for xy-position and z-position. In order to determine object location relative to the plane we use a weighted average algorithm:

\[
x = \frac{\sum_{i=1}^{n} x_i v_i}{\sum_{i=1}^{n} v_i}, \quad y = \frac{\sum_{i=1}^{n} y_i v_i}{\sum_{i=1}^{n} v_i}
\]

Resulting planar hand location (\(x, y\)) is calculated using the sums over sensor positions \((x_i, y_i)\) and sensor values \(v_i\) as weight. While there are different, more advanced localization algorithms, this approach has several advantages: The implementation is easy and computationally inexpensive, thus applicable on a low-power microcontroller. Furthermore it requires only a single sensor to work, different to other common approaches, including trilateration or multilateration that require at least three active receivers. For many hand positions, especially at the edge of the device less than three sensors detect an object.

Figure 8. Piecewise linear interpolation used to determine z-position of detected object

Since all sensor units are located on the same plane, a different approach is required to determine the z-position. The idea is to simply use the sum of all sensor values and associate it to height. Since sensor values increase when an object is approaching, this assumption is viable if the sensor distribution on the plane is sufficiently dense. Since the sum of values is increasing in non-linear fashion, we decided to use piecewise linear interpolation between fixed threshold levels, as seen in Figure 8, which delivered satisfying results.

Figure 9. Data processing virtual input device for modeled input device

Finally we noticed that the sensor devices are suffering from a considerable amount of noise, which required us to apply some interpolation on detected locations, to allow for smooth interaction. Two different instances of linear interpolation are running for planar position and height. The process results in the virtual data processing input device depicted in Figure 9, based on the notation detailed earlier.

4. PROTOTYPE

In order to test the various scenarios we needed a prototype that enables easy setup changes, while remaining portable. Therefore we decided to use sensors that are transmitting wireless to a base hub, and use wire as electrodes to experiment with different configurations. The disadvantage is that a dedicated power source is required, in case of our first prototype AAA batteries.

After scanning the market for available solutions we decided to base our prototype CY3271 PSoC First Touch Starter Kit by
Cypress Semiconductor that combines capacitive proximity sensing capabilities with wireless data transmission using a proprietary 2.4GHz protocol. Coiling the wire and placing it on acrylic tubes to get closer to the surface provided to be a good configuration with a good detection range.

Figure 10. Conceptual renderings and internal views of first (used for evaluation) and second prototype

This was the prototype used in our evaluation. However recently we have switched to a new iteration based on the same sensor type, yet with a more pleasing appearance and various modifications, like using a rechargeable battery for all sensors combined and switching to copper foil electrodes that provide a slightly improved signal-to-noise ratio. Both prototypes are displayed in Figure 10.

5. DEMONSTRATION APPLICATIONS

Regarding the general design of demonstration applications, the visual feedback is very important for this type of device. The lack of haptic feedback and direct visual presentation otherwise can make interaction difficult. All applications have therefore been developed using the XNA framework as overlay on Direct3D to freely design the user interfaces without operating system restrictions.

As described earlier we have determined three application scenarios in which our modeled input device shall be tested.

5.1. Unobtrusive media center control

Modern living rooms often feature many single-purpose devices, usually steered by individual infrared remote controls that have different layouts and often lead to confusion, especially for elderly users. In recent years media centers have appeared as alternative. These are devices based on PC hardware and powerful operating systems that try to provide access to all sources of media through a single interface. However interaction is still based on complex remote controls or keyboard and mouse.

Our vision for capacitive proximity measuring input devices in this context is the unobtrusive integration into living room furniture. The sensors work through any non-conductive material, thus are suited to be integrated into wooden or stone/ceramic furniture if the upper layer is not too thick. Our prototypes work with a rather thin acrylic layer on top; however tests have shown that it also works through wooden tables with a thickness of approximately 4cm.

Figure 11. Gesture-controlled image viewer demonstration application

A typical application for media centers is the display of images in a slideshow. Therefore we decided to implement this into our demonstration software with some common features enabled. As can be seen in Figure 11 the current position of the detected hand is depicted as green circle, with the path of previous locations displayed in fading circles providing instant visual feedback, thus allowing the user to perform supported gestures more easily.

The software iteration used is supporting four gestures:

- Switch to next image using a waving gesture from left to right
- Switch to previous image using a waving gesture from right to left
- Zooming into the picture if movement from bottom to top is detected
- Zooming out of the picture on movement from top to bottom

The gestures have two different modalities. The waving gestures are looking for a certain threshold to be exceeded and fire an event on occurrence. The moving gestures fire events until direction is changed or movement has stopped.

Recently we also have added a video viewer to the software suite that is using the same gestures for pausing and restarting the video and controlling volume instead of zooming.

5.2. Medical applications in sterile environment

The rise of medical image acquisition technologies have led to huge advances in modern hospitals, allowing more complex surgeries and increased chances for recovery. Modern MRT or PET devices deliver highly detailed 3D representations that are useful in planning the surgery but also occasionally have to be consulted in the ongoing procedure.

The benefit of using capacitive proximity sensing in this context is sterility. The surgeon is not allowed to touch any surface while operating on the patient to prevent infection. Thus if he is required to consult the 3D images taken before the operation he has to rely on an assistant which might lead to delays. A gestural interface based on proximity sensing would allow the surgeon to directly manipulate the images without having to risk sterility.
Therefore our demonstration for this application scenario is a 3D object viewer featuring a menu-based interface that is controlled by a combination of emulated point-and-click, as well as movement control. The common pan, rotate and zoom features of 3D object viewers are supported:

- Pan is a direct transposition of the hand movement onto the scene
- Rotate is creating an axis vertical to movement through the object central point and rotates relative to movement speed
- Zoom is working analogue to the zooming in the image viewer demonstration application

In order to increase the available screen estate the menu is initially hidden and has to be activated by moving the hand to the left side of the input device. Menu items are activated by holding the position on the item for a given amount of time. One additional feature allows switching between several preloaded models via menu entry.

5.3. Interaction device for the physically challenged

Certain diseases and health related problems may lead to difficulties in using common input devices like mouse and keyboard. Examples include carpal tunnel but also amputated hands or forearms. There are several input specialized input devices that target single aspects of this group.

Capacitive proximity sensor arrays can be applied very flexible. Changes in electrode design can lead to device with a geometry differing from our prototype. If that is combined with adjustments in gesture recognition algorithms it is possible to create devices based on the same technology that are suited for many different symptoms and diseases.

Our demonstration application in this scenario implements a screen lens for text and images. A regular feature in many operating systems that is suited for the elderly or otherwise visually impaired persons. We use it to demonstrate a direct location-based interaction. A regional zoom is applied at the current detected xy-position. The z-position determines the zoom factor; touching the prototype will lead to the highest magnification. Switching between documents or images is implemented using active regions. If the hand is held in the right region for a certain time the software switches to the next time. Likewise remaining on the left side will load the previous items.

6. EVALUATION

The evaluation was done with 18 persons, 12 male and 6 female with an age range between 20 and 50. Participants were invited from our research institute and nearby university and can be described as proficient in PC usage. We selected a group acquired with modern HCI technology, partially even working in UI research. The target of this evaluation was to get expert feedback, in order to optimize hard- and software as well as determining future steps in expanding this technology. Thus we associate the expected feedback to two groups:

- General performance of the device
- Comparing the different application scenarios and interaction paradigms

Each participant was given a five minute introduction into the features and interaction paradigms of each demonstration application. Afterwards they had ten minutes to test the program and later fill out a questionnaire about their experience. Besides acquiring general information the participants had to grade precision, how intuitive it is to use and if they could expect using it for long term purposes. Grades are ranging between 1 (worst) and 10 (best).

The results are displayed color-coded in the following bar chart, e.g. two participants graded precision with a 2, while three graded it with an 8.
The precision of the device did get the lowest grading. Participants could imagine using such a device for longer times, depending on the application scenario. Participants mostly liked using our prototype to control applications and managed to learn the required gestures in a short amount of time.

Regarding each application they had to grade their satisfaction analogue to general questioning between 1 (worst) and 10 (best).

The positive remark we gathered can be roughly put into three groups:

- The device very intuitive to use
- The idea of interacting this way is novel and interesting
- It is easy to control applications with those gestures

Likewise we identified three main groups for negative comments about the prototype:

- The device is not very precise
- The interaction speed is slow
- It can be tiring for the arm

6.1. Interpretation of results

Evaluation results on the general questions as well as analyzing the comments lead to an overall positive picture of the input device itself. Many of the negative aspects can be attributed to technical deficiencies of our prototype. Precision and interaction speed are the main concerns the participants had. We are optimistic that we can improve these factors by using different sensor and data transmission technologies and optimized antenna designs.

The most important negative aspect noticed in the evaluation is the tiring of the arm after using the device for prolonged periods. This factor is also affecting prospects for long-term usage negatively. We realize that this is a negative aspect inherent to the system that has to be taken into account when designing applications and user interfaces. The input device is suited best for scenarios where short interaction is followed by longer breaks, e.g. media center applications or presentation software. If that is not the case the user interface has to be designed in a way that the user does not need to interact for long periods. A first addition to our software made after the evaluation was to modify the screen lens application, where this prolonged interaction is most prevalent. We added what we call sticky lens; that is that the software is detecting when a user stays in a position and height for a certain time and sets the lens as stuck, meaning the lens parameters are stored and the user can remove his hand. As soon as the hand enters again in different position the lens is movable again.

A prevalent positive aspect in the evaluation is the intuitive use of the device. Provided with proper visual feedback, users were able to control the applications on their own quickly. They also considered such an interaction enjoyable due to its novelty. Similar reasons lead to the success of other gestural interfaces in the past few years. Based on this it is viable to further explore possibilities and limitation of input devices based on capacitive proximity sensing.

7. CONCLUSION

In this paper we have explored the design of input devices based on capacitive proximity sensors. We have presented a method to formally model input devices based on capacitive proximity sensors. In order to achieve this we have added a new singular input device to the modeling system of Card et al and developed a method to integrate the complex pre-processing that is occurring in many modern input devices.

To demonstrate this modeling we have formally described an input device based on six sensors that is able to detect the position of the user’s hand moving above it in three dimensions. In order to explore the performance and usability of this device we have built a prototype and created several demonstration applications showcasing typical features in medical context, media center control and interaction for persons with disabilities.

Finally we have done an initial evaluation with a group of expert users to explore the viability of our application scenarios and prototype, as well as getting feedback and ideas for future improvements.

The evaluation results suffered from technical deficiencies in the prototype but also offered a positive outlook on the potential acceptance and future of this category of non-haptic input devices.

Based on the evaluation results we have started to optimize hardware and software, resulting in a new prototype and additional features in our application suite.

8. FUTURE WORK

We are trying to further explore various aspects of the technology and user interface design for this type of device in future research.
We are currently working on a new iteration of the prototype that should have a much higher precision and increased interaction speed. Furthermore we want to explore user interfaces that are specifically designed for non-haptic interaction, supporting more gestures and having the ability to switch between various applications without having to fall back to traditional input devices like mouse and keyboard. In this context we also want to further explore the specific challenges of this interaction paradigm. E.g. we have noticed that it is sometimes difficult for a user to determine if his hand is currently in detection distance. Various approaches exist to determine a user’s intent that we want to explore. Finally we have also planned another evaluation, in which we want to compare the performance of several 3D input devices.

9. ACKNOWLEDGMENTS
We would like to thank all volunteers from the TU Darmstadt and Fraunhofer IGD Darmstadt that have been participating in our evaluation and provided helpful comments that will help us planning our future steps in researching novel interaction technologies.

10. REFERENCES