Rapid Prototyping of Customizable Physical User Interfaces by Means of RFID and Infrared Light

ABSTRACT ** (150 WORDS, WILL CHANGE AFTER IMPLEMENTATION)

Rapid prototyping of physical user interfaces (PUIs) can be very useful for designers if both the "look and feel" of a device and interaction with the system come together early in the design process [**Avrahami]. However, many toolkits including Phidgets [**Phidgets], iStuff [**iStuff], widget tapping [**widgettap], Switcharoo [**Avrahami], Calder [**Calder], and Lego's [**Lego] constrict the selection and amount of parts, thus limiting on the look and feel at the cost of easy interaction. I (**change to "we" later) plan on developing a system which allows designers to use their own parts, and use RFID and/or computer vision to allow these parts to interact with their system. In order to test the system, a user study will be run. The goal is to make the process of designing PUIs easier for the designer by enabling them to customize their parts.

Author Keywords

Rapid prototyping of physical user interfaces, customizable tools

ACM Classification Keywords

H5.2. Information interfaces and presentation (e.g., HCI): User Interfaces. – Prototyping.

INTRODUCTION ** (.5 - 1 PAGES)

In order to rapidly prototype and build physical user interfaces (PUIs), designers want to develop the "look and feel", or form, of a device and interaction with the system simultaneously [**Avrahami]. The reason for this is so the design process is more fluid, faster, and allows for more iteration [**BOXES]. Designers can make more practical designs keeping both form and interaction in mind. They are allowed to use their own devices which will allow them to envision their design and expand their creativity as they are prototyping. The design process of PUIs can be more efficient and robust because of the flexibility of possibilities at a low cost.

However, there is no current support that allows keeping form and interaction together throughout the entire design process. Building Objects for eXploring Executable Sketches (BOXES) [**BOXES] currently provides rapid prototyping of PUIs in the very early design stages. This is followed by the use of toolkits for a more sophisticated design, but not to the point where the designer is manufacturing their own parts. These toolkits including Phidgets [**Phidgets], iStuff [**iStuff], widget tapping [**widgettap], Switcharoo [**Avrahami], Calder [**Calder], and Lego's [**Lego] constrict the selection and amount of parts, thus limiting on the "look and feel" for the benefit of easy interaction. This limits designers to a certain possibility and quantity of devices, which isn't the goal when they are still trying to brainstorm endless options.

In this paper, we seek to bridge the gap between form and interaction in the earlier stages of design. We plan to improve on BOXES [**BOXES], by lengthening the maturity of the system. In the past, it has been useful for the very early stages of prototyping. BOXES will now support more input and output devices which are selected at the designer's discretion. Designer's can construct particular devices they wish to use, and specify this to BOXES so they test both form and interaction without inner working knowledge of the devices themselves. This will help the designer prototype until they want to actually manufacture the parts.

One way to simulate interaction of customized input devices is by using infrared (IR) light emitting diodes (LEDs). They can be placed on customized parts and can be simulated using computer vision. The LEDs can portably attach to devices (e.g. slider, knob), and will be read by an IR camera. If BOXES is programmed to look for a certain input device, it will look for certain events relating to the IR LEDS. For example, if the designer wanted to add a slider, the recognizer would look for the LED's position along a line to track how far the slider was moved. One issue with LEDs is occlusion, meaning if the LED is blocked BOXES won't be able to track it. We are presenting Radio Frequency Identification (RFID) technology to help aid in this issue.

RFID technology can be used to track devices that aren't in the line of sight. RFID tags can be placed on the input devices, and read using a small RFID reader. This is convenient because the RFID tags do not need power, only the reader does. The implementation is like the IR LEDs by having BOXES track certain events. If voltage level information can be read from an RFID tag, then we can essentially track the position of a moveable object, such as a slider or knob. However, an issue with RFID technology is the delay from reading multiple devices [**Paradiso]. A list of pros and cons of IR LEDs and RFID are shown in Table 1.

The rest of this paper is organized as follows: First, there will be a discussion of related work with respect to our system. The approach to building this system will then be discussed. An organized view of what will happen over the rest of the summer follows. In closing, resources still needed for the project will be listed followed by how this system should be evaluated.

RELATED WORK ** (1+ PAGES)

Input Devices

Input device possibilities have been investigated in order to provide designers with new ideas. Card et al. [**Card1] provided a way to describe input devices over a design space. They expressed devices in terms of position, force, and whether they were linear or rotary. In addition, they present different benchmarks for input devices: expressiveness (input conveys intended meaning), effectiveness (conveys meaning with ease), footprint (amount of space needed), and bandwidth (is the device is as good as the hand that is using it). When it comes to footprint, they found that some device's footprints don't increase as the screen size increases, such as light pen, touch pad, trackball, rotary pots, and joystick. However, this wasn't true for the mouse or tablet.

Tracking Methods	Pros	Cons
IR LEDs	Multiple LEDs can be used with no delay	Needs battery Occlusion
RFID	No battery Don't need line of sight	Slow reception to multiple devices

Table 1. These are listed tradeoffs between IR LEDs and RFID technology.

Card et al. added to their work by providing a morphological analysis of the design space presented beforehand [**Morphological]. This analysis used different input devices as points in a parametrically described design space. They refined their analysis mathematically on movement time and degree of difficulty for computer tasks using Fitts' law. They showed that different certain muscle groups, like the hands, allow for higher capability on the computer than others such as the arm or neck.

This design space of input devices might inspire designers to try out different types of inputs. This paper will build upon their work by providing the flexibility of different types of inputs designers can come up with using the design space. In addition, adding functionality for more touch related input and output devices are beneficial since they are the easiest and quickest to work with. This as a result could speed up the prototyping iterations.

Toolkits

In order to assist designers in providing functionality, many toolkits have been developed. Ayer's and Zeleznik's proposed the Lego Interface Toolkit which is a system that will allow rapid prototyping of 3D devices [**Lego]. Their medium was to use Lego blocks as an easily changeable and acquirable material. They found that their prototype widgets allowed for quick prototyping, but not for prolonged use.

Phidgets [**Phidgets], or physical widgets, were designed due to the fact that anytime a designer had to make a physical design where components are hooked together and need to interact with some form of low level programming; they needed to start from scratch. All of the low level coding is also fairly complex and can take quite some time to finish a project. The phidget allows people to spend more time on the physical interface and less time on the programming that isn't really the essence of the project. They paralleled the application of a widget to a GUI as a phidget is to a physical user interface.

A following work that focused on rapid prototyping of physical interactive devices was Switcharoo [**Avrahami]. Their work introduced the fundamentals of the letting a designer work on both form and interaction at the same time early in the design process. They used wireless input components that communicated via RFID and allowed designers to sketch the form of their interface. This allowed for rapid changes to the design. There was very positive informal feedback which confirmed the need for this type of work.

Further work was done with Phidgets and widget tapping [**widgettap] in order make customizable physical interfaces. Their work showed how to link physical interfaces to application widgets on the computer. This was done to link physical components from the Phidgets kit to software applications on the computer. There were many applications addressed for this system, one of them being a construction kit. However, they stated that external attachments are the bottleneck to customizable physical interfaces and that should be discussed more.

Another system which used various input and output devices to aid in rapid prototyping is iStuff [**iStuff]. They introduced many input and output sensors, and the ability to work across multiple users in a particular environment. They advertised iRoom, which could be part of the output (i.e. pressing a button to turn off the lights). It had an interfacing environment, called PatchPanel which can link input events using sensors or human input to various output applications already in existence.

Calder [**Calder] was a 3D toolkit to help aid in earlier design when it comes to physical interfaces. There were various inputs and output sources to customize the physical device and connect it to graphical user interface (GUI) applications for testing. These devices could be wireless and/or wired depending on design constraints with size and mobility. They assisted in the design process by adding the functionality one needs, while preserving flexibility.

More recent work was done with a wearable toolkit called the LilyPad Arduino [**LilyPad]. The LilyPad Arduino was a wearable electronic interface that people can program and integrate into clothing with conductive fabric and thread. They provided different types of sensors and software that people could use to program their device. This was targeted to hobbyists and students, with a slightly higher learning curve. Based on classes they held and presented in their paper it had a positive impact on design.

Our approach is similar in the fact that we want to support rapid prototyping of PUIs, but the main difference is that we want to have a (2D or 3D?) form that fits closer to what the designer is looking for. Want to allow for rotational and linear motion, push buttons, and a variety of outputs that may fit in the design space given previous work. Unlike toolkits, the designers will be able to provide their own easily accessible and changeable objects that they created. This is justified in Plywood Punk, where they say that designers should not be constrained to create new objects from a kit of parts [**Plywood]. In addition, we will make these devices wireless by portable battery with the IR LEDs, and make these easily replaceable for prolonged usage. RFID tags can be used for a long time since there is no battery. The main contribution is that we giving the prototype both higher fluidity (rapid prototyping) and fidelity (closer to desired form).

Prototyping Cycles

Some research exists to improve the designer's experience with rapidly prototyping PUIs. D.tools [**d.tools] was a tool that helped designers design, test and analyze their products. Their user interface showed the device, a flow chart between all of the user actions, and supporting source code. It allowed the designer to work with a physical prototype, meanwhile having it be simulated on the computer. This way a prototype can be simulated whether or not it is actually connected to the machine. In addition, D.tools provided videotaping which could be used to compare between different actions of a particular device or between many devices.

A more recent work emphasizes on a prototyping cycle of design, edit, and review. Exemplar [**Exemplar] was a tool that allowed designers to experiment with sensors and link certain events to application options with mouse clicks or movement. They allow for the cycle of demonstrate, review, and edit to be implemented so the designer can customize their input devices to their needs, dealing with single or multiple events or dealing with single or multiple sensors. Exemplar was an example of programming by demonstration because they were focusing on sensor based interactions.

While our work is helping in the prototyping design cycle, it is focusing more on the designing and testing. These tools could be used in conjunction with our work in order to have a better analysis across different implementations.

BOXES

The motivation behind BOXES [**BOXES] was the usefulness of linking a physical device with its application earlier in the design process. This emphasizes the ability to see form and interaction at the same time. This allowed for quicker iterations to improve the physical device. This can be used on existing software applications or someone's own developed software. They accomplish this by using common household items to create an early stage remote to interact with the computer via usb serial cable. This enables for clicking and keyboard events to be done with buttons which could control software programs. The buttons work with capacitive touch sensors, and the capacitance caused by the space between thumbtack and foil and possible human.

We plan to increase the maturity of BOXES so it can be used later in the design process. Our project builds on top of BOXES by allowing usage of simple everyday objects, and having an easy learning curve in order to rapidly prototype. Adding the ability to use people's own objects allows for a more creative design process.

THE APPROACH ** (WILL BE REMOVED LATER, OR GO SOMEWHERE)

My plan is to have a small processor, IR led, power hooked to each component. In order to quickly prototype my system, I am working with the Arduino Mini Pro processor.

I will be using the Arduino platform to program the chip. The IR led will have two uses: 1. Send a modulated IR signal to transmit information, 2. Allow an IR camera to read in the IR signal to allow for blob tracking. Right now, I am using Eyepatch [**Eyepatch] to work with blob detection by using their gesture tracking feature. In order to make this more robust, I will need to make an IR camera. I need to use a small battery that provides 3V and will be able to sustain 3.3V after using a converter board. Right now, the best option is a battery that is 12.5mm in diameter.

At this point in time, I have the Arduino board able to power down, and wait for an interrupt signal that can include a voltage rise or fall. This is useful because the board does not have to consume a lot of current, and the battery can recover until the next interrupt happens.

The goal is to have an arduino board and power source hooked to a particular device. I will have the current voltage hooked to the device (button or variable voltage), hooked to an interrupt in the board. When the interrupt triggers, I plan on reading the current voltage and transmitting that data via modulated IR signal using an IR led. In addition, the IR led will be read by blob tracking to assist in finding the position of the input device. This position will be used to actually implement what the user is trying to do on the computer or projected image.

PLAN FOR THE SUMMER ** (WILL BE REMOVED LATER)

7/22/2009 – start implementation of project

8/06/2009 – start planning user study

8/20/2009 – start user study

9/03/2009 – start writing paper

9/17/2009 – CHI paper due

RESOURCES STILL NEEDED FOR PROJECT ** (WILL BE REMOVED LATER)

I will be working with the Arduino Pro Mini board to assist in my implementation, with the goal of moving to a very small packaged board. This smaller board will not be needed until further into the implementation.

In order to implement multiple RFID tags, I need a multiple tag RFID reader. If I am working with IR modulated signals, I need to be able to get a reader to work with the signals.

In addition, I will still need to make an infrared camera so I can test blob detection with multiple IR light sources. Since my first attempt was unsuccessful, I need another web camera that better matches documentation online in order to make the process easier.

EVALUATION ** (WILL BE MORPHED INTO METHODOLOGY)

I plan to run a small user study with about 10 – 12 participants. I will have them do a task to design a physical user interface for some application, but I am still brainstorming what the specific task should be. Ideally, this would be a think aloud study so I can see how successful or unsuccessful this system is based on reactions throughout the process.

METHODOLOGY, RESULTS, DICUSSION, CONCLUSION ** (WILL COME LATER)

ACKNOWLEDGMENTS ** (WILL COME LATER AND BE HIDDEN)

Subsections

Sub-subsections

REFERENCES

1. [**Avrahami] Avrahami, D., Hudson, S.E. Forming interactivity: A tool for rapid prototyping of physical interface products. In *Proc. DIS 2002*, ACM Press (2002), 141-146.
2. [**Lego] Ayers, M., Zeleznik, R. The Lego interface toolkit. In *Proc. UIST 1996*, ACM Press (1996), 97-98.
3. [**iStuff] Ballagas, R., Ringel, M., Stone, M., Borchers, J. iStuff: A physical user interface toolkit for ubiquitous computing environments. In *Proc. CHI 2003*, ACM Press (2003), 537-544.
4. [**LilyPad] Buechley, L., Eisenberg, M., Catchen, J., Crockett, A. The lilypad arduino: Using computational textiles to investigate engagement, aesthetics, and diversity in computer science education. In *Proc. CHI 2008*, ACM Press (2008), 423-432.
5. [**Card1] Card, S.K., Mackinlay, J.D., Robertson, G.G. The design space of input devices. In *Proc. CHI 1990*, ACM Press (1990), 117-124.
6. [**Morphological] Card, S.K., Mackinlay, J.D., Robertson, G.G. A morphological analysis of the design space of input devices. *Proc. ACM Transactions on Information Systems*, (2), April 1991, 99-122.
7. [**widgettap] Greenberg, S., Boyle, M. Customizable physical interfaces for interacting with conventional applications. In *Proc. UIST 2002*, ACM Press (2002), 31-40.
8. [**Phidgets] Greenberg, S., Fitchett, C. Phidgets: Easy development of physical interfaces through physical widgets. In *Proc. UIST 2001*, ACM Press (2001), 209-218.
9. [**d.tools] Hartmann, B., Klemmer, S.R., Bernstein, M., Abdulla, L., Burr, B., Robinson-Mosher, A., Gee, J. Reflective physical prototyping through integrated design, test, and analysis. In *Proc. UIST 2006*, ACM Press (2006), 299-308.

10. [**Exemplar] Hartmann, B., Abdulla, L., Mittal, M., Klemmer, S.R. Authoring sensor-based interactions by demonstration with direct manipulation and pattern recognition. In *Proc. CHI 2007*, ACM Press (2007), 145-154.

11. [**BOXES] Hudson, S.E., Mankoff, J. Rapid construction of functioning physical interfaces from cardboard, thumbtacks, tin foil and masking tape. In *Proc. UIST 2006*, ACM Press (2006), 289-297.

12. [**Calder] Lee, J.C., Avrahami, D., Hudson, S.E., Forlizzi, J., Dietz, P.H., Leigh, D. The calder toolkit: Wired and wireless components for rapidly prototyping interactive devices. In *Proc. DIS 2004*, ACM Press (2004), 167-175.

13. [**Eyepatch] Maynes-Aminzade, D., Winograd, T., Igarashi, T. Eyepatch: Prototyping camera-based interaction through examples. In *Proc. UIST 2007*, ACM Press (2007), 33-42.

14. [**Paradiso] Paradiso, J., Hsiao, K. Swept-frequency, magnetically coupled resonant tags for realtime, continuous, multiparameter control. In *Proc. CHI 1999*, ACM Press (1999), 212-213.

15. [**Plywood] Schmitt, P., Seitinger, S. Plywood punk: A holistic approach to designing animated artifacts. In *Proc. TEI 2009*, ACM Press (2009), 123-126.

The columns on the last page should be of approximately equal length. ** (will be removed later)