

Sketch-a-TUI: Low Cost Prototyping of Tangible Interactions Using Cardboard and Conductive Ink

Alexander Wiethoff¹, Hanna Schneider¹,
Michael Rohs¹, Andreas Butz¹, Saul Greenberg²

¹University of Munich, Germany

{alexander.wiethoff, michael.rohs, andreas.butz}@ifi.lmu.de, hanna.d.schneider@googlemail.com

²University of Calgary, Canada
saul.greenberg@ucalgary.ca

ABSTRACT

Graspable tangibles are now being explored on the current generation of capacitive touch surfaces, such as the iPad and the Android tablet. Because the size and form factor is relatively new, early and low fidelity prototyping of these TUIs is crucial in getting the right design. The problem is that it is difficult for the average interaction designer to develop such physical prototypes. They require a substantial amount time and effort to physically model the tangibles, and expertise in electronics to instrument them. Thus prototyping is sometimes handed off to specialists, or is limited to only a few design iterations and alternative designs. Our solution contributes a low fidelity prototyping approach that is time and cost effective, and that requires no electronics knowledge. First, we supply non-specialists with cardboard forms to create tangibles. Second, we have them draw lines on it via conductive ink, which makes their objects recognizable by the capacitive touch screen. They can then apply routine programming to recognize these tangibles and thus iterate over various designs.

Author Keywords

Tangible user interface (TUI), prototyping, design process, low fidelity, graspable, capacitive touch surfaces.

ACM Classification Keywords H5.2 [Information interfaces and presentation]: User Interfaces – input devices and strategies, prototyping.

General Terms

Design, Human Factors

PROTOTYPING TANGIBLE USER INTERFACES

Tangible User Interfaces (TUIs) were conceived as a way to link digital bits with physical objects [2]. The basic idea is that people can interact with that digital information by grasping and manipulating physical things [5]. Our own interests are in designing interactive tangibles that work



Figure 1. Sketching conductive ink on a cardboard object creates a tangible object recognized by a capacitive surface.

with the new generation of capacitive surfaces, such as pads produced by Apple and the various products that run Android, and larger table surfaces implemented via capacitive vs. vision technologies. Early prototyping of such TUIs is critical. Unlike traditional GUIs, approaches, methods and idioms are still evolving.

Without early prototyping, it is far too easy to produce poor designs. Low-fidelity prototyping is especially important for considering a wide variety of designs, and then choosing what appears to be the most promising amongst them for further development. Buxton calls this 'Getting the right design, and then getting the design right' [1]. Low fidelity prototypes work because: they do not confront users too early in the design process with unimportant design details; they allow non-experts to participate in collaborative design sessions; and they invite high-level user testing [1]. This is why low-fidelity prototyping methods abound for GUIs [6], and are widely accepted and considered essential in the user-centered design process [4]. Surface vendors are not blind to the importance of prototyping. For instance, Microsoft Surface produces fiduciary tags that can be stuck onto any physical object. When placed on the surface, that tag, its position and its orientation are recognized and made available to the programmer. Unfortunately, these fiduciaries only work with vision-based systems, and won't work on capacitive surfaces. Consequently, developing physical prototypes for capacitive surfaces often requires a substantial amount expertise in electronics to instrument them [5,7], as well as time and effort to physically model the tangibles and the electronics they contain. Due to these difficulties, prototype development may be skipped, or handed off to 3rd party specialists (adding delays and costs).

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.

TEI 2012, Feb 19-22, 2012, Kingston, Ontario, Canada.

Copyright 2012 ACM 978-1-4503-0541-9/11/08-09....\$10.00.

The usual outcome is that the number of alternative designs considered and design iterations are severely restricted. Our goal is to develop a low fidelity (lo-fi) prototyping technique for TUIs that works with capacitive interactive surfaces. Physical objects would be the input device, while the surface would be the graphical and auditory output device for visuals. Our audience is designers or design teams that have some coding expertise, but no material, hardware or electronics expertise.

SKETCH-A-TUI: CONDUCTIVE INK ON PAPER OBJECTS

Our method, which we call Sketch-A-TUI, lets people rapidly construct lo-fi 3D paper objects that are recognized by a capacitive surface. For this to work, these paper objects must be easily created and tracked by the surface. Our only assumption is that people have purchased a conductive pen (see below), that they have a device with a capacitive surface such as a pad, and that they have some coding expertise on that device.

Tracking. Capacitive surfaces recognize changes in the capacitive field above the surface of the screen. Modern capacitive touch screens track these changes, and use that to sense the touch of one or multiple fingers. Using variants of this idea, others have tracked physical objects placed atop the capacitive surface, either through passive conductors [3] or active electronics embedded in the physical artifact [5,7]. Unfortunately, these methods are not appropriate for lo-fi prototyping by non-specialists, as they require knowledge of electronics to create the conductors. We contribute a new method that lets non-specialists create recognizable physical objects simply by drawing lines on them using conductive ink (see Fig. 1). Various commercial pens¹ are available that write conductive ink; they are usually used for repairing circuitry on a circuit board. We appropriate these pens to let end-users draw a conductive line (with a blob-like endpoint) from the top of an object to its bottom (Fig. 3); only the blobs are seen by the surface, as the lines are too thin to be recognized. When a person holds the object, his or her hand touches the blobs on the top. The conductive ink then transmits the body's capacitive load from the top of the tangible towards its bottom that is in contact with the surface. Thus the sensing grid of the capacitive screen device 'sees' the blobs at the end of the bottom lines as touch signals. We use multiple ink lines per object that lead to different blob-like end-points (see Figs. 1 & 3). Software can then differentiate between objects by taking the spatial configuration of the end points into account. In the simplest case, there are two end-points per object and the distance between the end-points identifies to the object. Based on the distance between these blobs (the touch points it sees) the device eventually produces corresponding forms of interactive behavior. Distinguishing between finger and TUI touches is done by checking if there is a constant distance between touch points (as

produced by the TUI) vs. a single touch point (as produced by a finger) vs. changing distances between multiple touch points (as produced in a pinch-to-zoom gesture).

Creating Tangibles. Another bottleneck in prototyping TUIs is the actual construction of the 3D objects. To mitigate this, Sketch-A-TUI provides prototypers with a variety of templates (Fig. 2) that lets them create various 3D shapes out of thin cardboard (e.g., cubes, pyramids, cylinders). Marks printed on the template show where differentiating conductive ink lines could be drawn (e.g., Fig. 1). The only skills required are cutting, folding, gluing and tracing lines. To create a tangible, the designer folds and glues the template into a shape, and then sketches the conductive ink marks on its outer shell to make recognizable by a capacitive screen. The only constraints are that the sketched conductive ink marks have to end at the side(s) of the object used to touch the surface (see Fig. 3), and that marks have to be above the minimum detectable distance and size of in order to be safely detectable. For example, different lines can be drawn on different sides, connecting to contacts with different distances: this means that different sides of an object can be distinguished as well. Alternately, the prototype can use three or more contact points per object to create recognizable 3D shapes that distinguishes between objects. Of course, designers are not restricted to these suggested conductive ink lines or the provided cardboard shapes. The presented technique works on any material on which conductive ink sticks. The advantage of paper is that the objects appear very generic, which speaks for their appropriateness as representations for TUIs in an early stage of development [1].

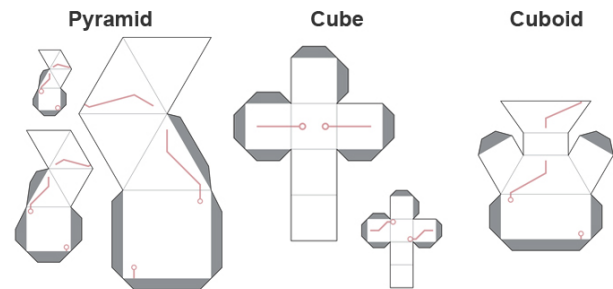


Figure 2. Example templates of paper objects. They collectively serve as inexpensive building blocks for creating a variety of TUI representations.

We currently offer 26 basic shapes as templates, all available online as a PDF that can be printed out on thin (0.5-1.5mm) cardboard [URL: <http://project-premium.org/sketch-a-tui/>]. Figure 2 illustrates three of the available template shapes in multiple sizes, while Figures 1 and 3 show the final product of the middle cube shape. This number can, of course, be expanded (perhaps by a DIY community) to include templates that define other shapes. Our templates serve as exemplars. Each provides clear instructions on how to create a shape, and where the

¹ e.g., Circuitworks' Conductive Pen Standard Tip, CW2200STP

conductive ink could be drawn to distinguish between them. Based on our own experiences with capacitive devices, the contacts on the bottom need to have a minimal size of about 5x5mm and minimal distance of about 5mm. This is because current devices are optimized for detecting fingers on the surface. The precise minimal values depend on the device used. An example of what acceptable contacts look like is shown in Figures 1 and 3. Overall, our approach makes it easy for an end-user to create and experiment with different physical forms, as well as different layouts of the conductive ink. Hence different forms of digital functionality can be “sketched” onto the cardboard artifact.

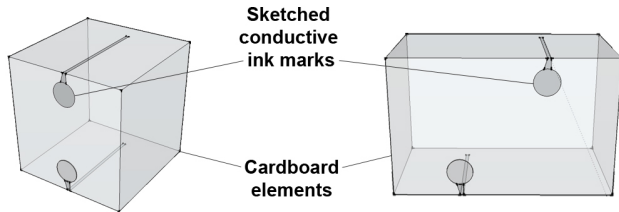


Figure 3. Through varying spatial configurations of contacts areas on the bottom of the cardboard elements it is possible to differentiate between them.

Coding. The final prototyping step does demand coding expertise, where the designer uses the programming or scripting environment of their choice to prototype interactive behaviors in response to the recognized objects and how they are being manipulated. Currently, we do not provide a software API for recognizing the touch points and/or for discriminating objects by the distance between those points, or the orientation of that object. This is because (a) we wished to remain agnostic to the actual device and its programming environment, and (b) it is a fairly routine programming exercise to recognize two touch events, measure the distance between them, use that to identify an object, and calculate the orientation of that object based on the angle of the connecting line relative to the surface. Still, an API to simplify even this step could easily be developed by ourselves or by others. In use, the cardboard objects can be positioned on any interactive screen with capacitive sensing technology allowing multiple touch points (e.g., Android tablet, Apple iPad). Using a range of development tools, the prototype can then design and test early functional behavior based on how these objects are manipulated by the end user; examples will be illustrated in the next section.

Overall benefits of Sketch-a-TUI. To sum up, we see the following benefits of our approach:

- low cost factor
- integration of digital functionality into passive paper-and-cardboard artifacts
- fast and easy reproducibility
- rapid way to prototype early explorations of tangible interactions

- no additional electronic components or batteries needed
- works with a variety of off-the-shelf technology (iPads, iPhones, Androids, etc.)
- enables collaborative design with teams including non-experts and end users.

EXPERIENCES

We tested and explored the technical feasibility of this approach. Our equipment was all off the shelf: a conductive pen (the Standard Tip CW2200STP by Circuitworks) to write conductive ink; standard 1.5 mm thick white non-coated cardboard for our tangibles; an Apple iPad (version 1) as the capacitive surface; and the XCode IDE for iOS as our coding platform. We created 5 basic objects from the templates (see Fig. 4, which shows them in use with an iPad), sized by the dimensions in Table 1. We then sketched lines of conductive ink onto the opposite sides of each of them as suggested by the template, each with different contact distances as listed in Table 1. We developed code that distinguished between these cardboard objects by the different spacing between the forwarded touch events, and associated every object with an identifying number.

Table 1. Specifications of our 5 shapes

Shape	Edge/ diameter	Height	Contact distance
Cube	40	40	35
Cylinder	35	40	30
Octagon	8	17	15
Pyramid	40	40	20
Octagon-on-Cube	10,28	40	25

We then implemented a simple software application, written in Objective-C in the XCode IDE, that calculated the distance between the two touch points and compared it with predefined ranges. Building one object took approximately 20 minutes. Occasionally, an object did not trigger the intended reaction; this was usually the result of inadequate ink. Repainting the conductive ink lines remedied these cases. In one of our test prototypes, a user holds one of the 5 objects on the iPad. A pie menu appears immediately underneath, where its size (between 3 and 6 cm in diameter) and color were unique for each of the 5 objects (Fig. 4). When the objects were rotated clockwise or anticlockwise on the screen the passed circular sector of the pie menu was highlighted yellow. In addition, the object ID and a number showing the object’s relative rotation direction when rotated were displayed on the screen. We also offered Sketch-A-TUI to a design team of a major company in the consumer electronics sector, who used it in the development process of a hybrid interface using tangibles on a capacitive screen (details cannot be provided due to non-disclosure agreements). The designers applied the proposed method and prototyped the TUI through many iterations. The resulting tangible interface design was so compelling that it will go into production in the near future, which speaks for the applicability of the lo-fi prototyping approach proposed here. We observed their design sessions

and interviewed the designers: their experiences were quite encouraging as the whole team highly appreciated the opportunity of creating early expressions of their ideas on the fly. Importantly, they stated that, prior to Sketch-A-TUI, they used non-interactive 3D renderings and mockups of their early ideas, which limited their internal design process.



Figure 4. A working Sketch-A-TUI prototype showing the graphics it is controlling on an iPad.

LIMITATIONS AND EXTENSIONS

Technical limitations include the fairly limited number of contact points that current multi-touch screens can distinguish at any one time. For example, the iPad we used can only distinguish 10 contact points at once. This limits the simultaneous presence of 5 objects, if each has 2 contact points. On the other hand, the small size of the iPad screen would make more than a few objects impractical anyway. Another limitation is that the touch points of the objects are only sensed while the user holds the object. This means that we cannot distinguish between an object lifted off the surface vs. just lifting the fingers off an object that remains on the surface. We are currently exploring whether these cases can be distinguished by analyzing the movement pattern of the contact points before lift-off.

Extensions. Different sides of a 3D cardboard object can be associated with different contact points on the bottom, so that different behavior can result depending on how the user touches an object. This is more flexible than a fixed one-to-one mapping of physical objects to virtual behavior. For example, grabbing the object in one place could activate a pie menu to select various options, whereas grabbing another part could activate a slider to continuously adjust some previously selected parameter. A complete interaction language could be defined in this way in a very simple manner. Additionally, objects could be assembled of simpler objects stacked atop of each other. The relative rotation or position of these sub-objects could activate different contact lines leading from one sub-object to another and eventually to contact points on the capacitive screen.

DISCUSSION

Our approach potentially allows an integrative co-design of physical and virtual interfaces while being technologically robust and reliable. Quick explorations of physical form factors and mappings of conductive ink lines to digital functionality enable the design team to quickly try and test early explorative concepts of their system on the fly. According to Buxton [1] a sketch of an early design concept should exhibit design properties such as minimal detail, quickness, disposability, inexpensiveness etc. Hence, we believe that a consolidation of low fidelity cardboard elements on the physical side and visually non-designed interactive interface representations on the digital side match well in an early phase of the design process and help in developing a TUI appropriately.

CONCLUSION AND FUTURE WORK

We presented a low cost approach to prototyping early explorations of tangible user interfaces that combines the rapid creation of physical 3D cardboard objects with drawing conductive traces that make those objects recognizable on current off-the-shelf capacitive touch screens. Because creating these objects requires little effort, our method allows designers to explore and produce more alternative designs, which help in getting the right design and the design right. We do recognize that our current instantiation requires programming. We are currently constructing a software toolkit that eases even this burden, and which will allow non-programmers to choose and link pre-defined interactive functionality to recognized objects.

ACKNOWLEDGEMENTS

We thank Alexander Kehr for his help along the project.

REFERENCES

1. Buxton, W. *Sketching User Experiences: Getting the design right and the right design.* Elsevier 2007.
2. Ishii, H., Ullmer, B. *Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms.* In *Proc. CHI'97*, 1997.
3. Kratz, S., Westermann, T., Rohs, M., Essl, G. *CapWidgets: Tangible widgets versus multi-touch controls on mobile devices.* In *Proc. CHI EA '11*, 2011.
4. Maguire M. *Methods to support human-centered design.* *International Journal of Human-Computer Studies.* 2001.
5. Patten, J., Recht, B., Ishii, H. *Audiopad: A tag-based interface for musical performance.* In *Proc. NIME '02*.
6. Snyder C. *Paper Prototyping.* Morgan Kaufmann. 2003.
7. Yu, N.-H., Chan, L.-W., Lau, S.-Y., Tsai, S.-S., Hsiao I.-C., Tsai, D.-J., Hsiao F.-I., Cheng, L.-P., Chen, M.Y., Huang, P., Hung, Y.-P. *TUIC: enabling tangible interaction on capacitive multi-touch displays.* In *Proc. CHI'11*, 2011.