An Exploration of Multi-Touch Display Systems

Christopher A. Wood and Anthony C. Barone
Engineering Methods of Software Usability

Abstract— *Human Computer Interaction, or HCI, is a constantly evolving field of study. As both hardware and software technologies continue to improve, the need for more efficient and usable HCI techniques increases. One of the most profound technological advances that has impacted HCI in recent years was the development of multi-touch display systems. The term multi-touch display system refers to a touchscreen display that has the ability simultaneously register two or more distinct positions of input touches from the user. Multi-touch display systems are the next step in natural user interfaces in that they allow the user to directly manipulate the system with theoretically no bounds on the level of physical interaction. In a way, this new sense of freedom is the beginning of a new era of computing in which multiple users can interact with each other simultaneously on a single system. Therefore, in order to continue forward with this technology it is essential to understand where it stands today.*

Keywords: User experience, Human Computer Interaction, HCI, Multi-touch.

1. Introduction

Human Computer Interaction, or HCI, is a constantly evolving field of study. As both hardware and software technologies continue to improve, the need for more efficient and usable HCI techniques increases. One of the most profound technological advances that has impacted HCI in recent years was the development of multi-touch display systems. The term multi-touch display system refers to a touchscreen display that has the ability simultaneously register two or more distinct positions of input touches from the user. This provides a foundation for a new genre of user interfaces (UIs) by giving users the ability to input more than command or action at a time. Multi-touch display systems are the next step in natural user interfaces in that they allow the user to directly manipulate the system with theoretically no bounds on the level of physical interaction. They have also begun to usher in a new era of user interaction and collaboration through the encouragement of multi-user systems.

The design, implementation, and usage of mutli-touch systems has changed dramatically in recent years. New low-cost designs and hardware components, along with enhanced software processing algorithms, have led to the production of highly efficient and scalable multi-touch display systems. Subsequently, these new systems have led to a shift in traditional user interfaces and user interaction gestures towards an entirely new user experience and genre of HCI. By making computer systems more user-friendly and native to use they have naturally attracted a new variety of users, most notably the elderly and disabled that previously struggled using computer systems.

The purpose of this paper is to explore the impact of multi-touch display systems on technology, the user experience, and users themselves. This exploration follows a bottom-up approach, starting with modern-day multi-touch display system design techniques and working towards the high-level user experience. We will explore many aspects of the design, implementation, and impact of multi-touch display systems as they exist today. Finally, we will finish with some final thoughts about our exploration.

2. Relevant Background

Before multi-touch technology emerged onto the market and into public hands, our interaction with computers and electronic devices occurred by means of commonly acknowledged peripherals: the mouse and keyboard. These components are quite effective in navigating and controlling a computer system and their life longevity has proven this. However, people have always felt an interaction barrier by only having control of one coordinate of location at a time. This is natural; our everyday real lives are filled with fluid and dexterous interactions requiring both hands and all fingers. Therefore, when people sit down at a computer, the interaction barrier that they feel is defined by the interruption of everyday fluid and dexterous motion.

Specialists and researchers of system usability began to see this barrier as a serious prevention towards completing certain tasks quickly and efficiently. Soon enough, more and more interest shifted towards technology that utilized a more natural user interface experience: touching. One of the first ever touch screen computing devices was the PLATO IV Touch Screen Terminal, which was developed in the Computer-based Education Research Laboratory at the University of Illinois in 1972 [1]. This device was a single-touch system that was designed as a computer assisted education device. Older touch screen systems like the PLATO IV tended to be pressure-based, meaning that input registration was determined by pressure on a given spot on the screen. However, as time went on and technology advanced, touch screens would start to appear on the commercial scene with point-of-sale systems, ATM machines, and airport check-in kiosks.

The idea of direct contact interaction with our fingers seemed to void the barrier people had dealt with since computers were made mainstream. Hence, more and more consumer electronics along with computer screens themselves would be developed as "touch-capable" and the trend took off. Early adaptations of touch technology allowed for only a single point of contact with the screen at any instant in time. While this certainly felt more natural to users opposed to interacting with a peripheral device, scientists and researchers begin to imagine the plausibility of sensing and processing multiple points of contact via multiple fingers simultaneously. Such multi-touch display systems have been around for a while now, dating back to the release of the Multi-Touch Tablet by the Input Research Group at the University of Toronto in 1985 [2]. This device was capable of detecting multiple locations of input, as well as the degree of touch for each registered input.

Unfortunately, at this time the multi-touch technology was still under heavy research. It wasn't until Jeff Han first presented the Frustrated Total Internal Reflection (FTIR) multi-touch surface at a TED talk in 2005 [3] that it really began to take off and catch the public's eye. Jeff Han is an HCI designer of FTIR-based multi-touch surfaces and founder of the start-up company called Perceptive Pixel, which is now the market's leading producer of advanced multi-touch systems. Ever since the first demo of FTIR multi-touch technology the interest in multi-touch systems has grown at an alarming rate and has continued to increase with the release of other ground-breaking devices such as the iPhone, Microsoft Surface, and Microsoft Wall [4]. In fact, the acceleration of new multi-touch technologies, applications, and user interfaces has seen a similar growth rate. This trend signifies the shift in hardware and software technologies from the traditional "point-and-click" to multi-touch systems that open up the door to a wide variety of applications.We will dive into this intriguing and ground-breaking technology of multi-touch, which ultimately will shape the next-generation of computer user collaboration and graphical interfacing through electronics.

3. Multi-Touch System Designs

Traditionally, multi-touch display systems have been produced using resistance based, capacitance based, and surface wave touch surface designs. While these multi-touch solutions may be effective, they are not necessarily cost effective. This is why designers of multi-touch systems have begun to focus on using optical-based implementation designs that are cheap and easy to implement, yet robust enough for large surfaces. Specifically, Frustrated Total Internal Reflection (FTIR), Diffused Illumination (DI), and Diffused Surface Illumination (DSI) are three of the most prominent optical-based implementation designs used to create cost-effective multi-touch display systems [4].

FTIR is an implementation technique adapted from biometric fingerprint image acquisition devices used to create multi-touch systems. It is unique in that it allows the acquisition of true touch information at high spatial and temporal resolutions, and is scalable to very large installations [5]. Use of the FTIR technique has been around since 1970 in biometric devices that detected the attenuation of light through a platen waveguide caused by a finger in contact, but have started to die off due to a lack of available machine vision hardware [3]. However, now that machine vision hardware (e.g. cameras and projectors) have become inexpensive and easy to integrate into custom solutions, more and more multi-touch systems are built using this technology. Infrared sensitive cameras and projectors are used to capture light waves that are reflected from the surface (as shown in Figure 1) in order to determine the position of objects on the projection surface.

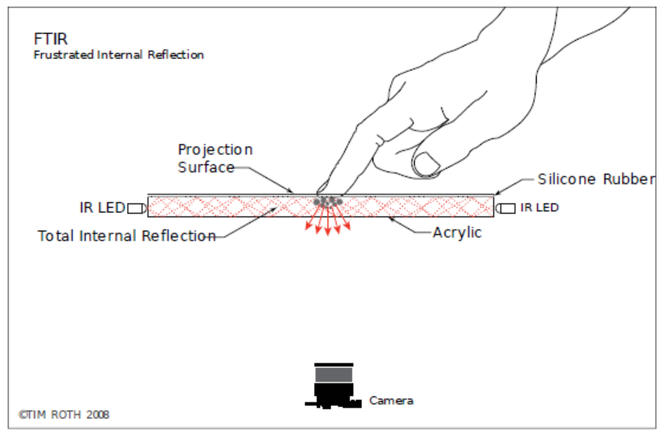

Fig. 1: General set-up of a FTIR system [4]

Diffuse Illumination (DI) is another optical-based image acquisition implementation design that is used to produce multi-touch display systems. The hardware implementation of DI systems is similar to that of FTIR in that they both require a projector and infrared sensitive camera placed behind a projection surface [4]. However, unlike FTIR systems, the infrared camera is placed behind the projection surface with the projector, as shown in Figure 2. The benefit of this implementation strategy is that the entire project surface is illuminated with infrared light waves, which means that even objects in close proximity to the surface are detected by the camera, rather than just those in direct contact with the surface. This allows for a much more robust recognition mechanism and gives software interface developers the opportunity to leverage this feature. However, while this feature provides much greater flexibility when implementing multi-touch applications, it does not provide an even distribution of infrared light waves across the project surface. The reason for this is that the cameras are located behind the surface at an angle, and thus the distance between intersection spots on the projection screen for infrared light waves varies based on the angle of the infrared cameras and their size. Most DI systems attempt to remedy this by using a couple infrared illumination sources.

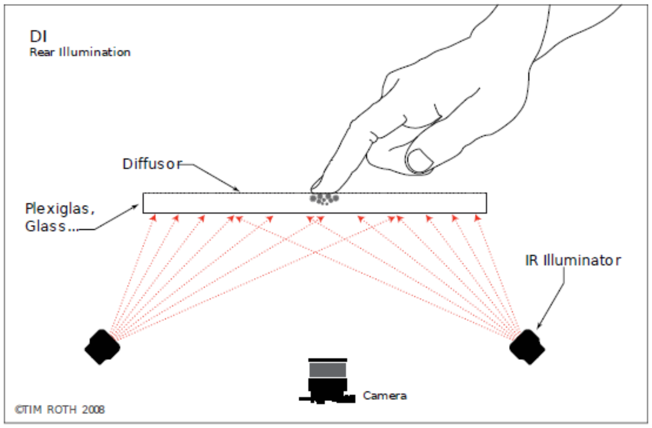

Fig. 2: General set-up of a DI system [4]

The last optical-based technique is Diffused Surface Illumination (DSI). What is unique with this implementation design is that it seeks to address the infrared light wave distribution problem in DI systems. It does this by employing the use of small particles that act as mirrors on the project surface that reflect and evenly distribute light waves as they hit the projection surface [6]. However, similar to the other optical-based image acquisition techniques, it does have its set of downfalls. Particularly, it has less contrast than typical DI systems due to the added “mirrors”, which ultimately means that there is the possibility of more ambient infrared because of this decreased amount of contrast. In addition, the size of systems fabricated using this technique is limited due to the physical properties of the project surface material.

All three of these designs share one major pitfall. That is, since they are all optical-based designs, harsh lighting conditions can effect the accuracy of the image acquisition and projection process. While they are still able to operate in such conditions, the results are undesireable since they are inaccurate. These system designs work best when placed in ideal lighting conditions where there is little interference that affects the infrared light waves as they move between the illuminators, the projection screen, and the cameras.

4. Hardware Components

When it comes time to actually fabricate multi-touch systems using one of the three optical-based designs described in the previous section, there are a variety of hardware components available for utilization. Each of these components are utilized differently in order to re-create the system and lighting effects necessary to capture user input on the projection screen. The general locations of each of

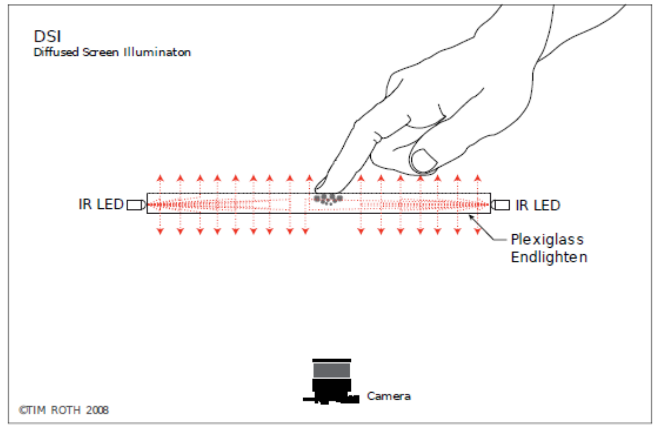

Fig. 3: General set-up of a DSI system [4]

these components are shown in Figures 1, 2, and 3 on the preceding pages.

In essence, each system will need some combination of infrared illumination sources, silicone compliant surfaces, projection screens, cameras, filters, and projectors. The infrared illumination sources are used to provide the infrared light source that covers the projection surface. Cameras are used to detect fingers in contact with the surface. Filters are used to control how much light reaches the sensor in the camera. Finally, projectors are used to present the actual image that users see on the project surface [4].

Depending on the design that is used during the fabrication process, these hardware components can vary in size, quantity, complexity, and most importantly, price. Much like any other business, trade-offs must be made between the cost of fabrication and the quality of the end product. Multi-touch system designers and companies that have the funds necessary to invest into new hardware components can do so to improve performance. However, trade-offs must be made for those that can’t afford to invest in new materials, and performance will suffer because of it. Some of these performance considerations are discussed in more detail in section 6.

5. Software Processing

In addition to these hardware components, software processing algorithms need to be utilized to support the image acquisition and projection screen update process on multi-touch systems. Similar to most graphics processing software solutions, multi-touch display systems employ a pipelined software processing algorithm that is used to incrementally transform a camera image acquired by the hardware components into a user interface event represented in software.

The purpose of this pipeline is to take raw image data as input and perform various transformations on it at each stage in the pipeline in order to produce input for the multi-touch application. For example, the software pipeline use for

FTIR multi-touch display systems has four different pipeline stages, including preprocessing, connected components, temporal correlation, and transformation, as shown in Figure 4. The preprocessing phase is responsible for removing any unchanged parts of the image data that remained constant from the most recent image data. Next, the connected components transformation is responsible for determining contact locations on the projection surface, which is followed immediately by the finding corresponding contact locations in different image data [7]. Finally, the camera coordinates are transformed to screen data in the post-processing phase. A similar pipeline architecture has also been implemented for DI systems, as shown in Figure 5.

Due to the complexity of the image translation data and sequential behavior of the transformation process, this pipeline architecture is well defined and structured in order to optimize its efficiency and throughput. It allows for multiple image data “packets” to be transformed simultaneously (in different stages of the pipeline of course). If the processing algorithm were implemented as a single stage “function” then the system performance would undoubtedly suffer if it was fed an excess amount of input data. The reason for this is that the queue that contains data “packets” to be processed by the function could reach a state where its growth exceeds the processing time of the function, thus causing a certain level of slack time or latency. However, by breaking up the processing into multiple stages based on their computational complexity and interaction with other components in the system, optimal performance can be achieved.

6. Performance Considerations

Although multi-touch design techniques, hardware components, and software processing algorithms have continued to improve in recent years, there are still performance issues that must be considered when producing such systems. These performance issues have been under investigation by designers and researchers ever since they were released. Consumers want systems that are responsive and don’t suffer from latency problems, and in order to appease them multi-touch display system designers must invest more money or make trade-offs during the fabrication phase based on several distinct factors that affect the product’s latency. For example, Apple felt the need for smooth interface interactions with their new iPad and decided to design a new processor especially for it. The A4 chip, with an increased clock speed of 1GHz, would support a “snappy interface” for not only quick system response time, but also smooth gesture animations. This is the fastest processor that Apple has placed in one of their handheld devices, topping the 600MHz (0.6GHz) processor of the iPhone 3GS [8]. Such an increase in speed, not to mention the hefty financial investment, makes it quite clear that the Apple strives to satisfy the strong

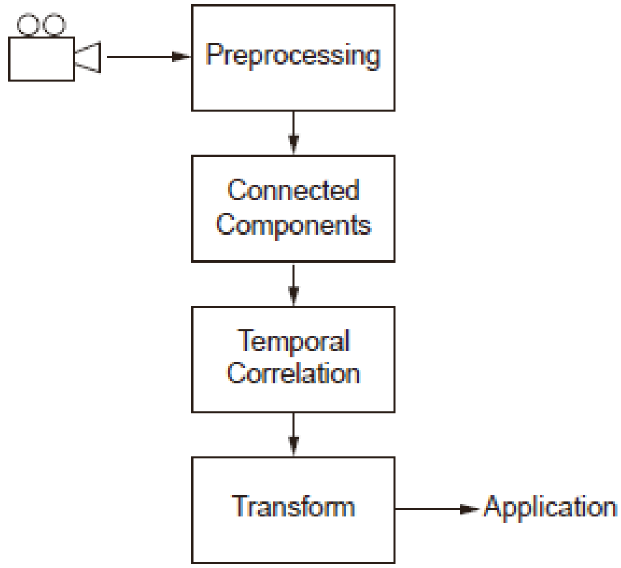

Fig. 4: FTIR Tracking Pipeline [4]

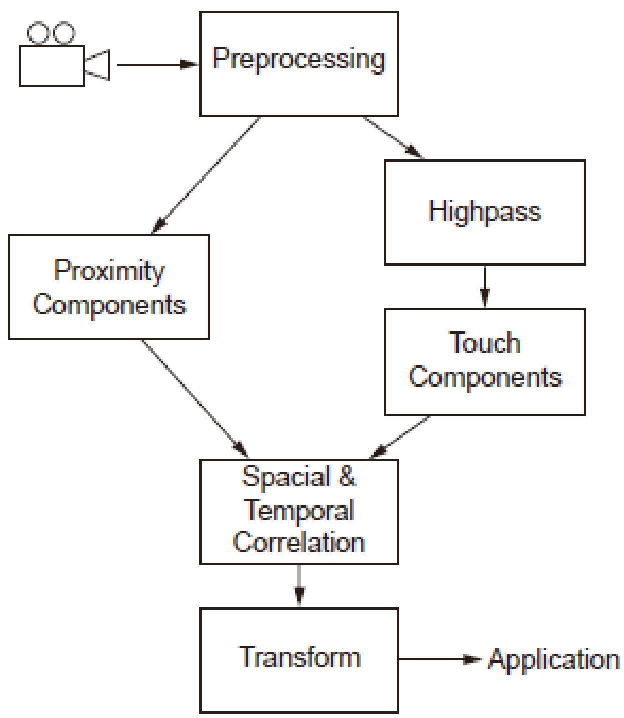

Fig. 5: DI Tracking Pipeline [4]

customer demand for seamless, real-time user interaction with their devices.

However, for those who aren't able to financially support the fabrication of new components to solve latency problems, trade-offs must be made based on the factors that actually effect the performance of these systems. Some of these factors are the system cameras, the software processing algorithm library, the actual application that is running on the system, and finally the projector. As you can see, each of these components correspond to a step in the system's operation cycle (similar to the fetch-execute cycle for a modern-day computer). Therefore, the highest level of optimization in these systems can only be achieved by optimizing each stage individually. If one stage has a less-than-optimal execution time then it will slow down the entire system.

The image capture and projection stages of the system operation cycle play a vital role in the system's performance. The camera is responsible for capturing image data and relaying it to the transformation component in the system, which is typically implemented in software. The projector is responsible for displaying image data on the projection screen for for the user to see. The latency time that exists within the capture and projection stages, such as the time between recording light and transmitting it from the camera sensors to the camera's embedded electronics for usage, must be minimized in order for these stages to achieve optimum efficiency. Multi-touch designers must consider the physical latency time that is inherent with these devices when designing their systems so as to avoid any unwanted bottlenecks that are caused by physical attributes of the hardware.

Perhaps the most computationally intensive stage in the system operation cycle is the image processing and transformation. As discussed earlier, a pipeline architecture is implemented in software to translate raw image data received by the cameras into user interaction events on the projection screen. The performance of this pipeline, which is often implemented and provided to the multi-touch system designers by a third-party, can be tested using standard software profiling techniques, the most useful of such being application counters (if the source code is provided). These counters can be used to measure the latency at each stage in the pipeline in order to make the necessary enhancements to the algorithm. In fact, as multi-touch systems continue to scale up in size and a software processing algorithm run on a standard microprocessor is not providing optimal performance results, multi-touch system designers should consider moving computationally intensive portions of the algorithm directly to application specific integrated circuits (ASICs) or even field programmable gate array (FPGA) boards to provide the necessary hardware acceleration. Regardless of its implementation, the performance of this algorithm must be carefull tuned for each application in order to achieve optimum results.

Finally, the software application that is running on the multi-touch display system is the next stage in the system's operation cycle. The purpose of the software application is to interpret the transformed image data that it receives from the multi-touch system, perform some internal computations and application logic, and then output new image data that is to be projected on the projection screen. The amount of time it takes to respond to new image data must be minimized in order to achieve full optimization of the entire system. Of course, in order to minimize this time the software application developers must implement an efficient and robust software solution that can handle real-time processing of image data in order to provide a responsive UI for the user.

7. Multi-Touch User Interfaces

Multi-touch display systems have also brought about a new era of software user interface mentality. Existing software interfaces that were designed without touch, or even with solely single-touch, capabilities had to be redesigned to account for the added functionality of multi-touch. For instance, the iPhone/iPad iOS and the Android Operating System both were implemented before any true deployment of multi-touch user interaction; however, the hardware devices associated with these platforms have evolved since their inception. With both systems now using multi-touch display hardware, proper modifications needed to be made to the UI code to react to a given set of gestures that were supported. Another example of this is the touchpad on most modern Apple and PC laptops. After adding multi-touch detection beneath the surface, new software drivers were developed to add a whole new way to perform common tasks, such as scrolling through a web page and browsing a photo album.

To complicate matters even more, not all devices are created equal and there is no "standard" hardware interface that we can base our software implementation on. The capability of a device varies between components, whether the multi-touch device is centered around camera systems or capacitive technologies. Apple's trackpad supports multiple gestures including a four simultaneous finger touch gesture, while HP's TouchSmart hardware is limited to capturing only up to two points of contact at a time. More large-scale devices such as the Microsoft Surface, on the other hand, is capable of recognizing and corresponding to over 50 points of contact [9]. Two important issues during the redesign of these operating system and application interfaces was the assurance of a "usable" system, and the adherence to existing usability standards that it complied with previously.

Stepping back and looking at the actual user interface from a design perspective, traditional shapes, links, and components will most likely need to be modified for multi-touch interactions. Unlike using a mouse, users have less pin-point accuracy when it comes to selecting components on the screen using solely their fingers. The contact area of a finger applying pressure on a multi-touch screen is rather large compared to the one-pixel precision of a mouse pointer, so objects on the screen should compensate for this, mainly by increasing in total area.

Furthermore, most legacy user interfaces emphasize on words, sounds, and pictures with little motion and movement to speak of. This design trend seems rather logical considering, up until recent years, most if not all devices required peripheral input from users either via mouse, keyboard, joystick, etc. Yet, our everyday lives are filled with fluid and dexterous interactions, so shouldn't our digital lives parallel this? With the introduction and growing popularity of multi-touch systems, it is increasingly more common to notice this new "layer" of UI design: fluid motion. Tomer Moscovich, from Brown University, conducted research featuring how multi-touch devices could make 2D computer prop animations much easier to perform [10]. The multi-touch user interface utilized and emphasized motion and movement, and Tomer discovered how "creating simple 2D animations by relying on [one's] natural sense of timing, and [one's] experience with real-world flexible objects" made it very easy for a beginner animator to produce a quality animation, contrary to traditional approaches of mouse-and-keyboard animating. The whole idea behind such a believable animation with this type of user interaction is the fact that all of the imperfections and nuances in the user's gestures to the computer prop are preserved, resulting in a more life-like presentation.

8. Multi-Touch Gestures

When considering the responsiveness, appearance, and feedback of an application, these all must be preserved when extending particular features to support multi-touch gestures. Popular gesture patterns that are emerging among most device manufacturers for interactions, such as multi-finger scrolling, zooming, rotating, panning, and image cropping [9].

Apple and Google both chose to incorporate the ability to modify the zoom level of a web page by simulating a "pinch" motion with two separate fingers (including the thumb) into various native applications. To zoom into the page, one must drag two different fingers away from each other, while both in contact with the screen. Zooming out is produced by the "pinch," or joining, two separate fingers, again while both are touching the screen (as shown in the top-right of Figure 6).

Considering the software implementation and usability standards of this multi-touch zoom effect, it is important for the browser itself to have a responsive, "lag-free" reaction to the user's finger-thumb gesture. By having a delay between the gesture and the interface reaction, there is a great risk of the user feeling "disconnected" from his or her actions, and therefore disconnected from the user interface itself. Parallel to this idea of responsiveness, the visual aspect of the user interface responding to these multi-touch gesture reactions

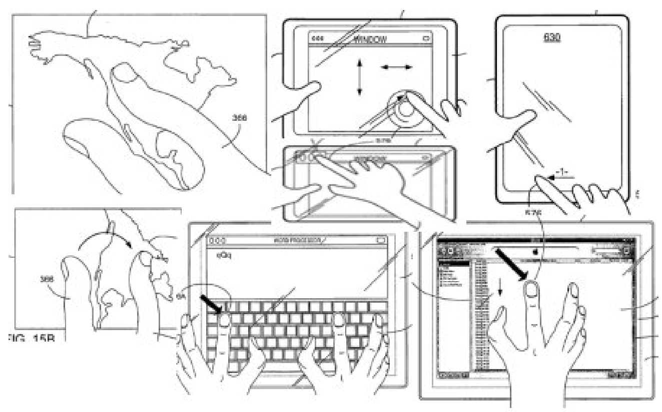

Fig. 6: A collection of multi-touch gestures.

is equally crucial. The ideal visual reaction to the "pinch" zoom gesture mentioned earlier would be a smooth page shrink/growth transition from the current zoom level to the desired zoom level, following the velocity and acceleration of the two-finger conjunction/separation.

Various other common tasks have been revamped with multi-touch software, such as typing using an on-screen keyboard. Before devices incorporated multi-touch hardware, it was quite difficult to type accurately and effectively with an on-screen "virtual" keyboard. The person would be required to completely lift his or her finger/thumb from the screen after touching a virtual key before the next input could be registered successfully. In addition, capitalizing a letter was not exactly intuitive, in that one could not just hold down the virtual SHIFT key with one finger and hit the desired letter with another finger as they would on a real keyboard. These usability issues were no longer pertinent after multi-touch display technology was integrated in, providing users the kind of functionality and responsiveness they would expect from a physical keyboard.

9. Impact on Technology User

Multi-touch display systems are a new type of HCI that bring with it a novel form of user experience. It has made a profound impact on how computer systems are used and who uses them. Traditionally, computers have always been used by one user at a time. However, with the release of multi-touch systems, computer system interaction has begun to gravitate towards a parallel interaction environment with simultaneous input from multiple users at a single time. One such example of this was the usability test that was conducted for CityWall, a large multi-touch display that was unveiled in a central location in Helsinki, Finland [11].

CityWall is a large-scale multi-touch display system that was designed to provide a user-friendly solution that supports 1) multiple hand tracking, 2) hand posture and gesture tracking, 3) high-resolution and high-frequency camera processing of up to 60 frames per second, and 4) computer-vision-based tracking that works in changing light conditions [11]. Over a time span of 8 days, it was observed that people began to work together when interacting with the device, creating an environment of teamwork, negotiation, and conflict.

Perhaps the most significant finding during this test was the emergence of parallel work between users. Although every user of the system was performing a similar task (moving pictures around in an application similar to Flikr), they were doing so simultaneously. This is very different from traditional computing where it was only possible for a single user to interact with a machine at a time. The idea of parallel work amongst users will usher in a new era of personal interaction and communication when using technology.

Multi-touch display system technology is even showing collaborative promise in classrooms with products such as the SMART Table. This table is the first multi-touch, multi-user interactive learning center that allows groups of early education students to work simultaneously on one intuitive surface [12]. Instead of giving a young student a computer to individually solve puzzles through single-user mouse interaction, the SMART Table introduces a variety of learning activities, puzzles, and games that do not exclude any individual from participation. Similar to the CityWall experiment, an environment of teamwork is established among children on account of multi-touch technology. Overall, there is recurring evidence that implications of multi-touch interfacing are a breakaway from how we understand the computer experience currently and a gateway for multi-user collaboration and interaction.

Another significant impact of multi-touch systems and their new user interfaces is the attention they are garnering from specific user groups, most notably elderly people. What is unique about this user group is that they are mostly uninterested in current technological advances unless they provide a much easier way of doing certain tasks. However, with the release of such user-friendly technology, elderly people are gradually beginning to gravitate towards use of multi-touch systems to incorporate them into their daily lives. For example, a multi-touch tabletop system called SharePic has been developed as a way to create a socially engaging way to manage digital pictures [13]. This system has been designed to account for the vision and motor problems that unfortunately effect elderly people. Speficically, it takes advantage of multi-touch gestures such as zooming, moving, and rotating to modify pictures that are on the screen. Since these gestures take little coordination and motor skills to complete, they can easily be performed by most elderly people, thus making it an easy-to-use tool.

SharePic is just one of the many examples of how multi-touch display systems have changed what people use technology. Another type of multi-user system built using a multi-touch display system is Microsoft Surface. This product is another tabletop multi-touch display system that gives users the ability to interact with applications using multi-touch gestures input [14]. However, this system is not limited to use for organizing digital photographs, as it has a wide variety of applications, including interactive and multi-user games and mapping applications. Another great example is the Apple iPad. As described earlier, it encompasses a multi-touch screen that is capable of a wide variety of gestures. Since this device is extremely usable and portable, it attracts almost every age group. It is designed to for ease of use by virtually anyone with its simple user interface, large on-screen keyboard, and size modification options [8].

In addition to the aiding the elderly with daily computing tasks, multi-touch systems are also capable of aiding the visually impaired. Most touch screen devices are largely inaccessible to blind users due to their inability to visually locate objects on the screen. However, researchers at the University of Washington have created “Slide Rule”, which is a way to utilize multi-touch display systems in conjunction with an audio-based interaction scheme that enables blind users to access touch screen applications [15]. The purpose of Slide Rule is to provide a completely non-visual interface that uses a touch screen as a “talking” touch-sensitive surface. There are four distinct multi-touch gestures utilized by this application to interact with the system. They are 1) a one-finger scan to browse lists, 2) a second-finger tap to select items, 3) a multi-directional flick gesture to perform additional actions, and 4) an L-select gesture to browse hierarchical information. These gestures are shown in more detail in Figure 7.

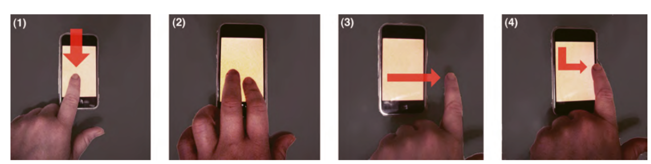

Fig. 7: Slide Rule uses multi-touch gestures to interact with applications [15]

What is unique about Slide Rule is that the multi-touch gestures are tailored to specific applications on the device that the user is interacting with. For instance, if the user is using the phone application then a second-finger tap gesture is the equivalent of calling a contact. However, if the user is using a media player application then a second-finger tap gesture is the equivalent of playing a song. By keeping the generic description of these gestures the same among all applications on a device the user can easily adapt to and use them without any visual input.

10. Final Thoughts

Prior to diving into this research exploration project we both found multi-touch systems very interesting. As computer programmers we rely on traditional "point-and-click" computing for most of our daily computing tasks. However, we both own and use multi-touch smart phones on a regular basis. Our interest in this subject was fostered by these devices, and throughout our exploration this interest has continued to grow. Perhaps the most notable findings we discovered during our research was the simplicty of modern-day multi-touch display system designs and their impact on technology users. Since we are both Software Engineering students we did not have a great deal of knowledge about the implementation details behind multi-touch display systems. However, after exploring the the different techniques such as FTIR, DI, and DSI optical-based implementation, we now understand what is happening behind the scenes.

The simplicity of these designs in terms of the physical hardware components necessary to construct such a product was astonishing. It seems as though, given enough time to let Moore's Law turn today's hardware into smaller, faster components, large-scale multi-touch surfaces will be entirely possible and found everywhere. It is even plausible to consider the possibility of interactive 3D environments given the appropriate advancements in lighting and mirror technology.

The other interesting aspect we found during our research was the impact of multi-touch systems on technology users. Not only do multi-touch display systems encourage multi-user computing, but it actually creates a social environment in which users can function. Unlike traditional computing where it was often difficult for multiple users to work together at a single workstation, multi-touch display systems foster interaction among users in a collaborative setting. In a sense, this can be thought of as an entirely new era of computing. As a society we are gradually shifting away from traditional "point-and-click" computing and moving towards multi-user interaction computing, where the proverb "two heads are better than one" really comes into play.

We also found it interesting that multi-touch display systems have such a profound impact on those who use technology. Considering the fact that multi-touch display systems and user interfaces are the next step towards an entirely natural user interface, we originally thought that people would be less accepting of this new type of interface. However, instead of resisting change, more and more people have welcomed the technological advancements brought about by multi-touch display systems with open arms, including those people that have never used such technology before in their lives. This evidence supports the claim that multi-user computing through multi-touch display systems is truly a new genre of computing that will continue to gain speed as we move into the future.

New and exciting ideas that revolve around this multi-touch phenomenon are emerging, from re-inventing the way we use computers to controlling robots with our fingertips [16]. This technology is even making its way into the classroom with products such as the SMART Table. Researchers are finding new ways to interpret touch interactions and to process higher amount of simultaneous contact points, thus opening a whole new range of applicability to various domains. Complementing this new technology spectrum, advancements in processing power and hardware are consequently decreasing latency times of multi-touch machines for a more real-time interaction. In addition, improved software algorithms for various animations, transitional effects, and image processing allow for a more usable and aesthetically pleasing experience. We can only imagine what the future will hold regarding the advancement of multi-touch technology, possibly expanding from device and table surfaces to 3-dimensional objects and beyond.

References

[1] B. Buxton, "Multi-touch overview," http://www.billbuxton.com/multitouchOverview.html. Accessed on October 27, 2010.

[2] ——, "A multi-touch three dimensional touch-sensitive tablet," in *Proceedings of the 1985 Conference on Human Factors in Computer Systems, CHI '85*, San Francisco, April 1985, pp. 21–26.

[3] J. Y. Han, "Low-cost multi-touch sensing through frustrated total internal reflection," in *Proceedings of the 18th annual ACM symposium on User interface software and technology*, ser. UIST '05. New York, NY, USA: ACM, 2005, pp. 115–118. [Online]. Available: http://doi.acm.org/10.1145/1095034.1095054

[4] J. Schoning, P. Brandl, F. Daiber, F. Echtler, O. Hilliges, J. Hook, M. Lochtefeld, N. Motamedi, L. Muller, P. Olivier, T. Roth, and U. von Zadow, "Multi-touch surfaces: A technical guide," 2008, technical Report TUM-10833.

[5] J. Han, "Multi-touch sensing through frustrated total internal reflection," Website, accessed on October 30, 2010.

[6] T. Roth, "Dsi - diffused surface illumination," Website, accessed on October 30, 2010.

[7] Y. Han and R. Wagner, "An efficient and fast parallel-connected component algorithm," *Journal of the ACM (JACM)*, vol. 37, no. 3, pp. 626–642, 1990.

[8] Apple, "ipad," http://www.apple.com/ipad/. Accessed on October 30, 2010.

[9] D. Wigdor and G. Morrison, "Designing user interfaces for multi-touch and surface-gesture devices," in *Proceedings of the 28th of the international conference extended abstracts on Human factors in computing systems*, ser. CHI EA '10. New York, NY, USA: ACM, 2010, pp. 3193–3196. [Online]. Available: http://doi.acm.org/10.1145/1753846.1753954

[10] T. Moscovich, "Multi-touch interaction," in *CHI '06 extended abstracts on Human factors in computing systems*, ser. CHI '06. New York, NY, USA: ACM, 2006, pp. 1775–1778. [Online]. Available: http://doi.acm.org/10.1145/1125451.1125786

[11] P. Peltonen, E. Kurvinen, A. Salovaara, G. Jacucci, T. Ilmonen, J. Evans, A. Oulasvirta, and P. Saarikko, "It's mine, don't touch!: interactions at a large multi-touch display in a city centre," in *Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, ser. CHI '08. New York, NY, USA: ACM, 2008, pp. 1285–1294. [Online]. Available: http://doi.acm.org/10.1145/1357054.1357255

[12] P. Steurer and M. B. Srivastava, "System design of smart table," in *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications*, ser. PERCOM '03. Washington, DC, USA: IEEE Computer Society, 2003, pp. 473–. [Online]. Available: http://portal.acm.org/citation.cfm?id=826025.826357

[13] T. Apted, J. Kay, and A. Quigley, "Tabletop sharing of digital photographs for the elderly," in *Proceedings of the SIGCHI conference on Human Factors in computing systems*, ser. CHI '06. New York, NY, USA: ACM, 2006, pp. 781–790. [Online]. Available: http://doi.acm.org/10.1145/1124772.1124887

[14] Microsoft, "Microsoft surface," http://www.microsoft.com/surface/en/us/default.aspx. Accessed on October 30, 2010.

[15] S. K. Kane, J. P. Bigham, and J. O. Wobbrock, "Slide rule: making mobile touch screens accessible to blind people using multi-touch interaction techniques," in *Proceedings of the 10th international ACM SIGACCESS conference on Computers and accessibility*, ser. Assets '08. New York, NY, USA: ACM, 2008, pp. 73–80. [Online]. Available: http://doi.acm.org/10.1145/1414471.1414487

[16] M. Micire, M. Desai, A. Courtemanche, K. M. Tsui, and H. A. Yanco, "Analysis of natural gestures for controlling robot teams on multi-touch tabletop surfaces," in *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*, ser. ITS '09. New York, NY, USA: ACM, 2009, pp. 41–48. [Online]. Available: http://doi.acm.org/10.1145/1731903.1731912