SCIMP Final Report

CMPE 123A
March 18, 2009.

Prepared for review by
Kip Laws
David Munday
Stephen Petersen

Team Members
Jas Condley
Eddie Izumoto
Kevin Nelson
Matt Thrailkill
Zach Walker

Abstract

SCIMP (Santa Cruz Interactive Multi-touch Platform) proposes to simplify the collaborative work flow on a computer by creating a large, multi-user interface built into the surface of a table. This interface will allow for touch input from up to six people simultaneously while presenting high-resolution video output on the same surface. The collaborative nature of such a system goes beyond what can be accomplished with a typical computer console. Ultimately this allows increased productivity.

Introduction

When most people hear the word "architect", they typically think of a person sitting at a drafting table, sketching designs which they then bring into CAD applications to render simulations. Computer architects deal with hardware description languages, which are often verbose and unintuitive. With the popularization of inexpensive multi-touch human-computer interfaces, it has become apparent that computer architects no longer need to be limited by the semantics of a programming language, nor the inefficiencies of pen and paper based design.

SCIMP will draw upon existing multi-touch table research to develop a scalable platform capable of tracking input from up to six users while maintaining a high standard of display resolution. The design will be applicable to many disciplines, the first SCIMP implementation will reside in the Micro Architecture Santa Cruz (MASC) lab where researchers will develop advanced, intuitive, and collaborative CAD tools for processor development.

While the theory behind multi-touch technology has been around for some time, only recently has commodity processor technology become powerful enough to support affordable implementation of these techniques. Although implementations have largely remained hobbyist projects, several commercial platforms have been built: The Microsoft's Surface[1]and a platform from Perceptive Pixel[2]. Neither system is currently available to the general public.. Although much of the research behind Perceptive Pixel is available at Jeff Han's NYU website[3], no new research has been posted since Perceptive Pixel was founded.

The NUI Group (Natural User Interface) was founded to fill this gap in research and make multi-touch more widely available among open-source applications. As of this writing, their contributions are libraries and tools to simplify the creation of a multi-touch display as well as a growing community of touch-table research for a variety of hardware.

SCIMP aims to move forward the research base by developing techniques for scaling multi-touch systems to larger sizes. These techniques involve supporting multiple displays, multiple cameras, and accelerating the touch detection and gesture recognition algorithms using graphics processing units (GPUs). The project will also aim to expand the gesture library currently available by developing a method for training the computer to recognize new gestures. The training procedure will involve applying the gesture rather than writing code, making it easy for non-programmers to create new gestures. Additionally, the project will continue research into touch surfaces, experimenting with a frustrated total internal reflection (FTIR) scheme using a tempered glass surface.

Support for the project has been graciously provided by Professor Jose Renau and Corning Glass.

---

1. "Microsoft Surface." Microsoft Corp. http://www.microsoft.com/SURFACE/Default.aspx, accessed March 9th, 2009.
2. "Perceptive Pixel." Perceptive Pixel. http://www.perceptivepixel.com/, accessed March 9th, 2009.
3. "Multi-Touch Interaction Research." New York University. http://cs.nyu.edu/~jhan/ftirtouch/index.html, accessed March 9th, 2009.

Proposed Objectives

The project can be broken into the following phases of implementation, illustrated in Figure 1:

- Table construction
- FTIR implementation
- Camera interface
- Image processing
- TUIO interface
- Projector display correction

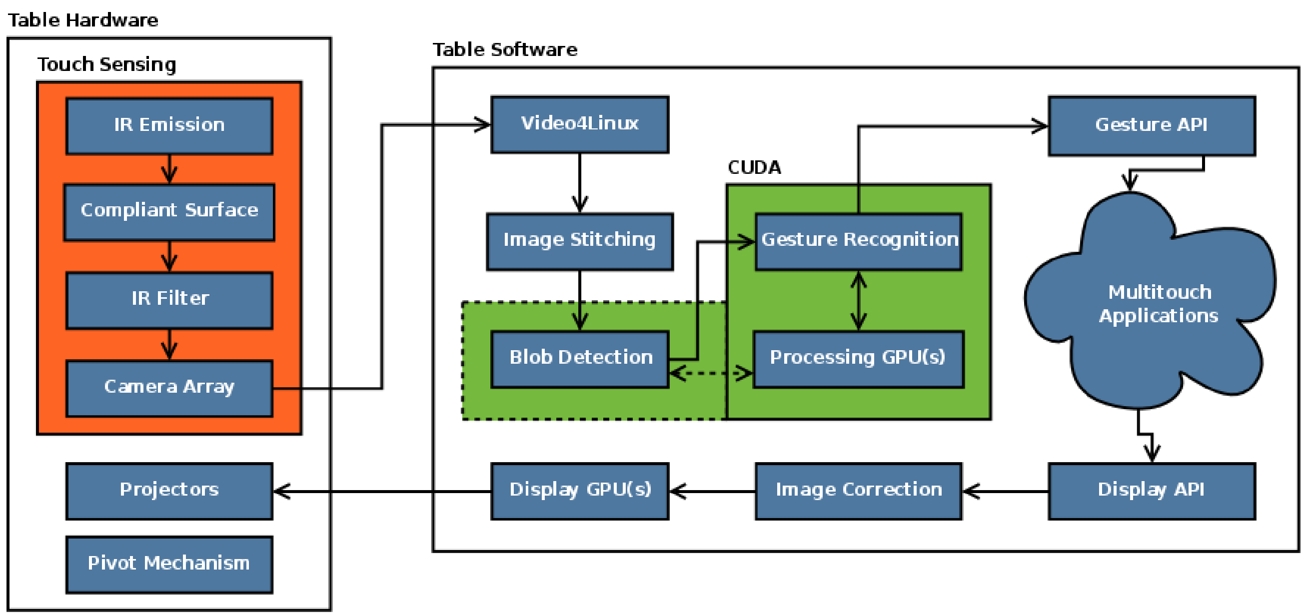

Figure 1 - Project block diagram

The goals the team has set forth are:

- Construct a 4' x 6' table capable of tilting 60 degrees
- Develop of a Frustrated Total Internal Reflection (FTIR) based touch sensing solution using infrared light
- Construct an FTIR-compliant touch surface
- Implement an array of web cameras modified to capture infrared touch points from the FTIR system
- Build A high-performance, CUDA-capable PC to handle multi-touch processing, image processing, and run user-level applications
- Implement an array of projectors for projecting to a 4' x 6' surface at 50 DPI
- Build upon OpenCV and other existing open source touch recognition software
- Create custom software interfaces to handle scalability
- Improve gesture library integration.
- Implement the TUIO protocol in the gesture API
- Develop software DSP for image processing.

Time Allocation

Task Name	Duration	Start	Finish
Prototyping	7 days	Mon 3/30/09	Tue 4/7/09
Camera Array Hardware	3 days	Thu 4/9/09	Mon 4/13/09
Computer Hardware	8 days	Thu 4/9/09	Mon 4/20/09
Table Structure	9 days	Thu 4/9/09	Tue 4/21/09
Stitching Software	19 days	Wed 4/15/09	Thu 5/7/09
Touch Detection	18 days	Wed 4/22/09	Wed 5/13/09
Display Handler	22 days	Wed 4/22/09	Tue 5/19/09
Gesture Recognition	24 days	Wed 4/22/09	Thu 5/21/09
IR Emission system	7.5 days	Thu 4/23/09	Sat 5/2/09
Compliant Surface	9 days	Sun 5/3/09	Wed 5/13/09
Debugging	13 days	Sat 5/23/09	Tue 6/9/09

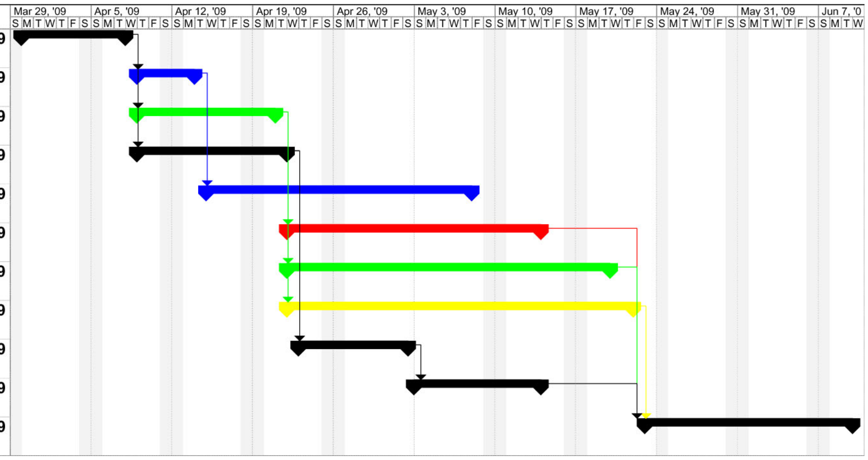

Figure 2 - Abridged Gantt Chart

The Gantt chart in Figure 2 clearly displays the dependency of the entire project on the critical first-week prototype. The prototype will be a single projector multi-touch table much like the ones that have been implemented already. This allows for the software development to ramp up more quickly without waiting for the completion of the larger, more complex table. A complete Gantt chart is included in Appendix A.

Intended Approach

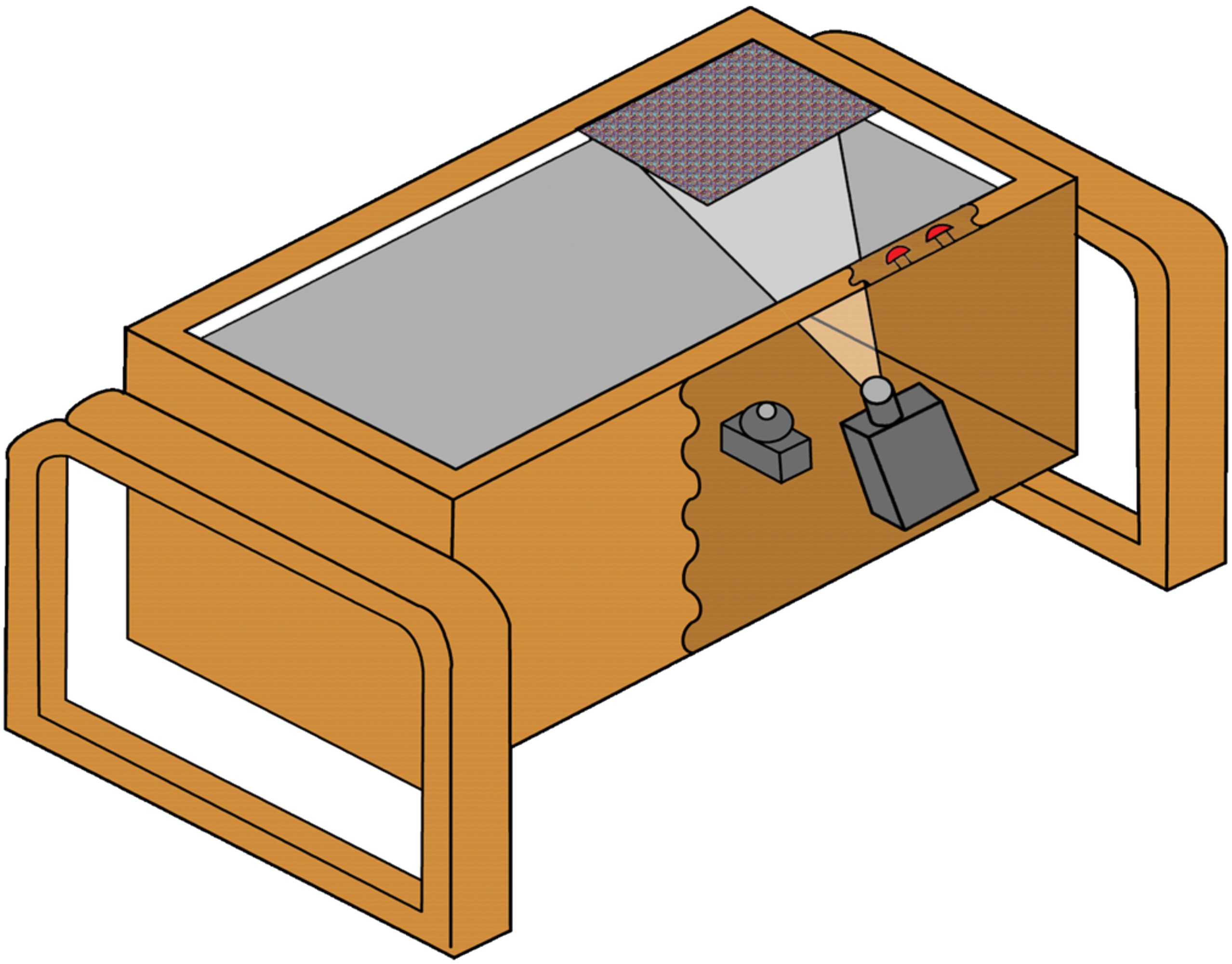

Figure 3 - Table Mock-Up

As shown in figure 3, the top of the table will hold a glass surface. Infrared light will be injected into the the edges of the glass by an array of infrared emitters such that the light is trapped inside due to total internal reflection. Touch input works by deflection of infrared light off of a finger pressed against glass. This is more commonly known as FTIR image-sensing, which takes advantage of the difference in indices of refraction of glass, air, and skin. When an object touches the glass, the refractive index is changed at that point and infrared light will escape from the glass. Some of this light will go into the interior of the table, where it will be "seen" by the array of webcams. Normally, webcams come with IR filters installed for various reasons. These filters will be removed in order to capture the infrared from the emitters. Additionally, IR bandpass filters will be installed in order to simplify image processing.

The webcams are connected via USB to a host computer running Linux. In Linux, Video4Linux (V4L) is the standard application programming interface (API) for capturing from video sources. A program will be written which will use V4L to take the video feeds from the individual webcams and reconstitute them as a single video stream. This final video stream will be a capture of IR touch information.

A second program will be written which consumes this video stream and produces touch information. It will apply image processing techniques to isolate the blobs in frames which are caused by finger presses that have deflected IR out of the glass. It is unknown at this point what the exact implementation will be, but we are planning to use the OpenCV library to help process the video frames. A low pass filter will be used to eliminate the Gaussian noise. The resulting image of this operation will only have the touch blobs. Detecting the position and size of the blobs in this image is straightforward after the aforementioned processing. An example of this detection is shown in Figure 4 below.

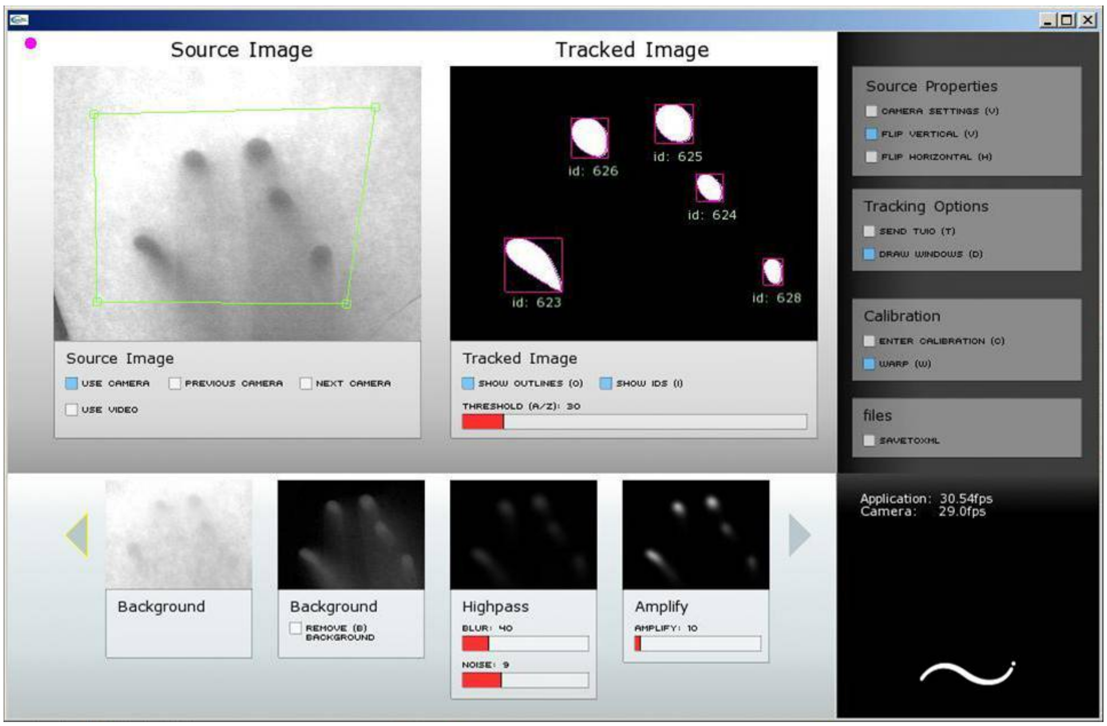

Figure 4 - Touch Sensing

CUDA is NVIDIA's architecture for writing applications for their GPU hardware. Because the GPUs can perform many operations simultaneously in the hardware, tasks that share similar computing of a large data set are good candidates for running on a GPU. NVIDIA has an FFT library for CUDA which may be suitable for accelerating the low pass filtering and any other DSP work which arises. The blob detection may be suitable for CUDA as well because algorithmically it involves looking at many pixels simultaneously. David Munday has begun work on a CUDA blob-detection program, but it is unclear whether it's performance will be an improvement.

CUDA has performance drawbacks such as latency involved in transferring data to the card and in programs which are not massively parallel in their workload. At this point it is unknown whether the performance gain can offset the latency of data transfers to the card or whether the touch and gesture processing will be suitably parallel.

Touches over time represent gestures. Therefore, the blobs will be considered over lengths of time in order to determine what gestures, if any, have been made. At this stage of planning, it is unclear if this work will be done in a third program or if it will be subsumed by the image processor.

An interface will be written which will serve these gesture events to any interested applications via the TUIO protocol over UDP. TUIO is a standard protocol for communicating generic gesture events, and it is commonly run over UDP. Using TUIO in this way allows us to run pre-existing multi-touch applications with little modification as well as possibly coordinating gestures over a network in the future.

In the host computer, there will be three video cards with two video outputs each. The outputs will be connected to six projectors which are mounted inside the table in order to project onto the underside of the glass surface. An IR-transparent layer will be mounted on the underside of the glass in order to provide a projection surface. The projectors themselves will each display video at an 800x600 pixel resolution. This low resolution was chosen in order to minimize price and maximize lifetime. Expected lifetime of the bulbs is approximately 30,000 hours. Six projectors will provide for a total resolution of 2400x1200. In order to eliminate gaps between the projected images, the projectors will be mounted such that their projections slightly overlap. Because of this, the individual video outputs will have corrections applied in OpenGL to trim their display areas appropriately and also to correct for possible skewing.

The GPUs in these video cards may also be used to accelerate the image processing mentioned earlier, pending performance evaluation.

The table body will be constructed to allow pivoting up to a 60 degree angle, as well as operation as a horizontal surface. The interior of the table will house all necessary components of the table being the projectors, the computer hardware, web cams, and necessary power routing, rotating with the table and keeping the total number of moving parts to a minimum.

Division of Labor

Jas Condley is a Computer Engineer with an Electrical Engineering minor and mechanical background. He has experience designing, building, and debugging embedded, high-speed systems and constructing human-computer interfaces. He is also the current president of Tau Beta Pi and organized the group for the SCIMP project. As project leader he will be in charge administration of the team, and the budget. He will also be the primary architect of the table, including mounting and routing of electronics and power internally. He will also be constructing the compliant surface and FTIR interface.

Eddie Izumoto is a Computer Engineer and has systems and parallel architecture background. He has experience programming on parallel architecture platforms, and building mechatronic systems. He will be the primary architect of the parallel image processing implementation in CUDA as well as the implementer of the image processing libraries.

Kevin Nelson is a Computer Engineer with an Electrical Engineering minor and an Electronic Music minor. He has experience building human-computer interfaces and a strong mathematical background. He is also the current president of the IEEE. He will be assisting in the administrative tasks and reassembling image data from the webcams for touch-point detection.

Matt Thrailkill is a Computer Engineer with a Computer Science minor. He has experience developing Linux-based software and OpenGL applications. He will be working with Video4Linux for image processing and implementing the display API with image correction. He will also be the main integrator of hardware with Linux drivers.

Zach Walker is a Computer Engineer with strong digital signal processing background. He has experience writing DSP software and human-computer interaction background, and writing sockets-based applications. He will be integrating present gesture libraries to include additional gestures and assisting with the design of the DSP for image processing.

Work Completed

As of this writing, the extent of our completed work is limited to research, projector quality assurance and basic web camera tests to validate Linux driver functionality. The chosen projector should meet all necessary requirements of brightness and resolution. Tests so far using the Playstation Eye camera seem to have a significant lag between image capture and availability to the operating system.

Budget

Item	**Cost**
Table hardware costs:	
Projectors	$5000
Glass	$300
Silicone + hardener	$125
Digiline Contrast Projection Surface	$600
Building Materials	$325
<u>Subtotal:</u>	$6350
FTIR implementation costs:	
Web Cameras	$150
IR LED apparatus	$350
Bezel	$150
Discrete Parts	$75
Filters	$100
<u>Subtotal:</u>	$825
Total:	$7175

Table 1 - Budget

The projectors were chosen to be relatively inexpensive while maintaining lifespan. The choice here was primarily between lifespan and resolution. To minimize cost over time of the platform LED projectors were chosen due to their long bulb lifespan, estimated at about 8 years of consistent research.

Glass was chosen over acrylic due to the ability to resist scratching and overall durability. Silicone and hardener create what is known as a compliant surface, which allow a greater amount of infrared LED to be shined downward.

The projection surface was considered for its grain. Since a large number of dots per inch are required, a projection surface with a fine grain was necessary. Reviewing research already conducted, the Digiline projection surface was chosen.

At present a wood structure has been chosen, at an estimated cost of $325. This includes mounting materials, but does not include tools for working with the wood.

The FTIR implementation is estimated based on relatively inexpensive cameras and IR filters. It includes the infrared LED's for prototyping as well as the final implementation and index-matching epoxy to transfer the maximum amount of power into the glass. A bezel along with various discrete parts to improve visual appearance are also included in this budget.

Donations

In addition to the budget listed in table 1, donations of a computer with three x16 PCI Express slots have been made, as well as 3 dual DVI-out video cards for driving the projectors. Glass samples will be donated by Corning glass, with additional support in the form of consulting.

Constraints

The project is constrained by three factors: client specifications, target cost (and funding), and products currently available on the market. Per the client's specifications, the first iteration of the table must maintain a display quality of 50 pixels (dots) per inch (DPI) and support a minimum resolution of 2700x1400. As mentioned above, we will only be able to hit 2400x1200 due to the current LED projector availability at our target price point. We have, however, discussed this with the client and come up with a clearer set of priorities. We will use with LED projectors due to their lifespan, but our area of display will be smaller than the area of the table surface. This design decision means that, as projector technology advances, the client will be able to upgrade the projectors inside the table, meeting the minimum specified resolution and display quality, without changing the physical structure of the table.

Conclusion

With these elements in place, we can build a table which could theoretically scale to arbitrary sizes. As an open platform, it will foster a new wave of innovation in user interface design and collaboration. Following the standards that are already in place for a touch platform, we ensure interoperability with future applications and continued viability as a research device. The platform will be usable for a wide variety of people from the musician to the human-computer interaction researcher to our motivating client, the computer architect.