Proteus Project: A Tangible User Interface Based on Computer Vision and Multi-touch Research

Submitted to the Engineering Excellence Fund Committee
University of Colorado at Boulder

February 28, 2008

Eric Baer
Computer Science
EricBaer2@gmail.com
(901) 497-6710

Eric Baer
Computer Science
EricBaer2@gmail.com
(901) 497-6710

Daniel Delany
Computer Science
Daniel.Delany@colorado.edu
(970) 309-8598

Dr. Dan Connors
ECES
dconnors@colorado.edu
(303) 735-7199

Dr. John Drumheller
Music Technology
drumhell@colorado.edu
(303) 735-0272

Dr. Mike Eisenberg
Computer Science
duck@cs.colorado.edu
(303) 735-0272

Abstract

The Proteus Project is an innovative tangible human-computer interface that allows the user to manipulate physical objects on a table's surface to accomplish tasks within a virtual environment. The Proteus will support extensive multi-touch interaction with capabilities of over 100 simultaneous points of contact, as well as recognizing the position and orientation of any object tagged with a unique computer-recognizable marker called a fiducial symbol. Raw input data is captured by an infrared camera underneath the table, and processed by an open source computer vision algorithm to be used for the control of applications running on the internal computer. Visual feedback is provided through a graphical user interface projected onto the surface from below. The prototype application for this platform will allow the user to control and generate real time audio signals in a simple, intuitive way. The long-term goals include the development of other intuitive applications, as well as an API to promote simple and customizable integration into classroom environments. The Proteus represents a new paradigm in human computer interaction by providing a way to abstract the idea of a computer from the creative process.

Project Description

The Proteus Project is an independent study research project for the spring 2008, through spring 2009 semesters based on computer vision and multi-touch research. The project has the support of multiple faculty members in both the School of Engineering and School of Music. Our vision is to create a tangible user interface that is powerful and data-rich while remaining intuitive enough that a child could learn its basic controls without instruction. By developing a user interface that has orders of magnitude more bandwidth (information transfer between the user and computer) than a standard mouse and keyboard, we intend to abstract the notion of a computer from the task at hand and encourage more creative freedom using common digital devices. The initial application of this control surface will be as a musical instrument and composition device. As the project continues beyond the prototype stage, we plan to refine the device as a new computing platform. We hope that this project will allow development of other exciting applications such as intuitive multi-touch control for programs like photo manipulation suites and computer-aided drafting environments, as well as tools for data visualization and manipulation. This will all be possible by creating a general purpose API (Applications Programming Interface) to allow for more flexible applications development within our research, and as a development environment for teachers and outside developers who wish to design applications beyond the initial scope of the project.

Proteus Capabilities

The Proteus concept is an interactive surface computing environment controlled by multi-touch input and by any object with a unique, computer-recognizable marker called a fiducial symbol printed on its surface. The device will be a self-contained tool that has the ability to as a black box requiring little or no configuration beyond loading an application. The table can detect over 100 simultaneous points of contact as well as the position and orientation of dozens of objects on the table. In addition to the general user interface, the table is capable of live audio signal processing. This includes two microphone channels (+48V phantom power), and six line level inputs for instrument microphones, guitars, keyboards etc.

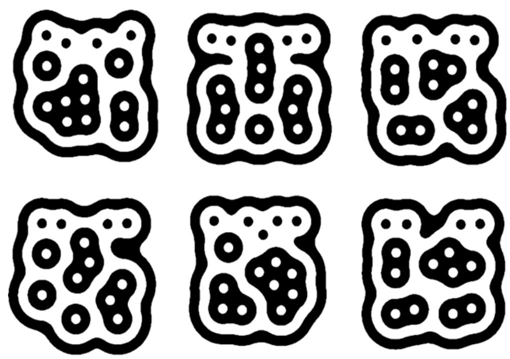

Figure 1: Example of Reactivision Fiducial Symbols

Hardware/Software

The basic design consists of a translucent surface with a near infrared camera placed underneath to capture an image of the surface 36 times per second. This camera is connected to a central computer running an open source computer vision framework called Reactivision. The Reactivision engine recognizes points of contact and fiducial symbols in the camera image and outputs data regarding their position and orientation using the TUIO data protocol, a protocol designed for this purpose, onto UDP port 3333 for use by external applications. This data is piped directly into Max/MSP, our sound generation engine, and is used to generate music according to rules programmed in a Max/MSP patch. The device's Graphical User Interface (GUI) will be programmed in Jitter, an extension of Max/MSP which allows seamless integration between Max/MSP patches and graphic visualizations. The GUI will be projected onto the surface from below using a short-throw projector allowing for a large projection within the strict size constraints of the table height. In the second iteration of the Proteus, an API will provide a layer of abstraction between the computer vision engine and the application layer.

Max/MSP, the project's signal processing and sound-generation tool, is a graphical development environment for music and multimedia used by musicians and software designers in its many forms for over fifteen years. The software includes an API, encouraging development of external programs like ours, and is widely regarded as the "lingua franca" for developing interactive audio programs and multimedia software, making it an ideal choice for an emerging technology. An educational license and a history of student support are both important factors in this decision. Finally, Professor John Drumheller, one of the supporting faculty members, has extensive experience programming in Max/MSP, and will be aiding in the creation of the prototype.

One difficulty with this design is the fact that the table must be able to have an image projected on it without oversaturating the camera's sensor. This is accounted for by using both the visible light spectrum and the near infrared light spectrum for the projector and the camera respectively. Since the projector projects entirely within the visible spectrum, the Proteus uses infrared irradiation to illuminate the surface, and a camera with an infrared pass filter to remove all visible frequencies. From the camera's point of view, the projected image is invisible. This design scheme was chosen because it is extremely robust, relatively inexpensive, and highly scalable. Another important design constraint is the throw length of the projector, the minimum distance it can project a focused image of a given size, and the viewing angle of the camera. The impact of these factors can be limited by using a mirror system to reflect the camera's and/or projector's images around the inside of the device to extend the throw length needed beyond the table's longest dimension.

Prototype Application

The prototype application is a musical instrument and composition tool geared towards live performance. It will act as a blank canvas on which sounds are generated and manipulated using sets of physical objects placed on the table. The objects will be divided into a few categories: generator, sample, filter, control, and container. The first two object types, generator and sample, will serve as the two methods of generating sound with the Proteus, while the filter, control and container objects will allow the user to extend control over the way sound or input is generated.

- Generator objects will include each of the eight elementary waveforms (sinusoid, saw, square, triangle, pulse, white noise, pink noise, and brown noise) and will function as the basis of any synthesis done with the table. Using these objects together in various combinations can create virtually any sound by a process called additive synthesis.
- Sample objects are user-created and can contain any sound of any length. They have the ability to capture output from the Proteus, as well as sounds from an external device such as a turntable, guitar or MP3 Player.
- Filter objects will be a collection of simple effects such as reverb, echo, chorus, and distortion. These objects will affect other signal objects in their proximity.
- Control objects allow for dynamic control and automation of concrete object parameters. As an example, a control object may be set to pan an instrument left and right at every second measure of a song, or to apply reverb to only a particular pattern during the chorus.
- Container objects can be placed on the surface to "save" information generated by a collection of objects. The sound created by all of the objects within the user-defined radius of the control object can be saved and abstracted out as a single object, clearing the work area and removing clutter from the project.

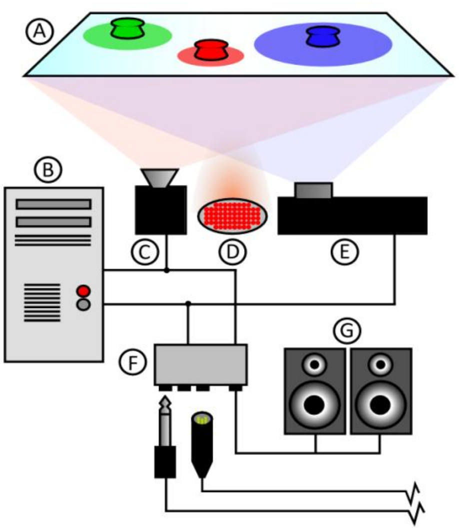

Figure 2: Device Schematic
A - Surface; B - PC; C - Camera; D - IR Irradiator; E - Projector; F - Audio Interface; G - Speakers

These objects encompass the bulk of the tools important to electronic music production and will allow the creation of infinitely complex and musically interesting compositions. In addition to the basic objects, there will be a few specialized objects such as an input object which allow the user to manipulate audio coming into the Proteus in real-time.

Each object type will share a unique but similar set of controls based on two control functions: the object's orientation on the table and its internal parameters. The orientation control will be common to all objects and will allow the user to control the object's primary parameter by rotating it on the table as if it were a knob. Each object will additionally have a radius that can be expanded by placing a finger on the object's edges and dragging outward. This expandable radius will be the foundation of the Proteus zoomable interface. As the radius of an object increases, touch-sensitive controls such as dials, sliders and switches will appear and become modifiable within this radius. These controls will allow the user to modify the object's internal parameters where the sensitivity of control is proportional to the objects radius. Parameter availability and zoomable control is experimental in the confines of this project. While the goal is to create an advanced, highly controllable tool, we must consider that the more control we give, the less intuitive the Proteus has the potential to become. Considerable thought must be devoted to this aspect of the design to ensure a balance between ease-of-use and flexibility.

Importance of Research

As a musical tool, this promises to be a giant leap from point-and-click music composition into the arena of live performance. A musician will not have to focus on the technicalities of the music and is free to be creative within this simple but powerful interface. Computing is now at a point where, for many tasks, availability of processing power is a moot point. From the standpoint of a computer scientist, the question of "what can we make the computer do?" has been replaced with "how can we make the power of a computer accessible and useful?" A program like Adobe Photoshop is incredibly powerful image processing software but the interface is tedious and the controls make simple tasks like blending, shading and drawing difficult using the mouse and keyboard interface. Mundane tasks such as visualizing a computer's file systems or information and simply moving files and folders is unnecessarily tiresome. The fundamental flaw with these systems is that they assume the "virtual hand" of the mouse to be as nimble and intuitive a tool as your own physical hands. The importance of this research is to move computers away from the antiquated idea of a black box with a screen, keyboard, and mouse towards a new archetype of intuitive, direct human-computer interaction. As with any new technology it is hard to see how far reaching the implications of the technology will be. We hope that by creating a robust hardware platform and a simple API, we will be able to give the capable faculty and student body of the School of Engineering and Applied Science a springboard for the incredible and creative applications of the future.

Impact of Research

The Proteus project aims to give back to both the University of Colorado at Boulder, particularly the School of Engineering and Applied Science, by presenting the finished product to the Engineering school faculty as a teaching aid. The Proteus promises to provide an engaging, visual, and tactile way to teach difficult abstract concepts such as a representation of data structures or visualization of molecule interactions. This projects plans to facilitate the integration of the Proteus into engineering classrooms through the use of its API for easy creation of class specific applications. We hope to enable teachers to write reusable and customized educational programs for the device that suit the teaching styles and needs of various classroom settings. Also, by making the system a portable it could be brought to any classroom to run these programs with relative ease. Computer Science Professor Michael Main has shown interest in using the Proteus to explain data structures in a tactile environment for a future CSCI 2270 courses. Electrical and Computer Engineering Professor Dan Connors has discussed with us the possibility of installing a device in the Computers as Components Laboratory for use in future ECEN 2120 courses. Here again, we will not try and imagine the possibilities the Proteus has for the University of Colorado faculty, we only wish to provide them with an invaluable resource to use at their disposal.

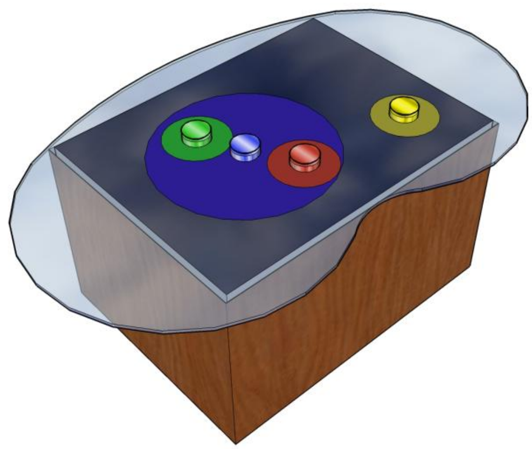

Figure 3: Concept Drawing of Completed Device

Because of the portability of the Proteus, it can also take the form of a temporary/permanent installation. As an installation, the Proteus could provide the general student body and the general public of a potentially attention grabbing example of the University's research efforts. During informal talks with various faculty members of the College of Engineering and Applied Science, the Proteus has been a proposed technology demonstration piece at the engineering advising fair. As a permanent installation, the Proteus could serve as another practical demonstration of technology with a broad base of appeal.

Timeline

The general timeline of our project is in two general phases. The first phase of the project is our proof-of-concept phase – to be completed by the end of the spring 2008 semester. Here we will demonstrate the feasibility and usefulness of the Proteus by designing and building a basic prototype and developing the prototype's music generation application. In this phase we will have the freedom to test different hardware layouts, and work around unforeseen development obstacles in a low budget bare-bones environment. Phase one is completely funded by a combination of personal investment and a contribution made by Professor Dan Connors. In this part of the project, we will spend a great deal of time testing different interface designs, conducting user tests, and polling faculty and students about what they would like to see as part of the next iteration of Proteus. Phase one is broken into three specific goals: general design and development research, hardware construction, and the design of the prototype application. We currently have completed the general design and development research, and are working on the hardware construction. We plan to have the hardware built and fully functional by the end of March 2008. Beginning April 1 2008, all project time will be devoted to creating a music application for the table and refining its controls. The completion of this portion of the project will coincide with the last day of the spring 2008 semester.

Phase two of the project will begin in the fall semester of 2008 and will be the concrete implementation phase. Here we have the enormous advantage of a second build. We will have met and overcome many challenges from the first phase and can plan for a much smoother hardware construction time, as well as an optimal layout and more robust design. In phase two of the project we also have the advantage of higher quality materials such as a camera with a better resolution and frame rate. We hope that the phase two build will allow us better object resolution and much closer to real time tracking. Phase two will be divided into four main sections: construction of hardware, development of an API, a revision of the prototype application for the API, and applications research. While some factors such as shipping times and the possibility of bad parts are out of our hands, we plan to finish the phase two hardware construction no later than October 1, 2008. Though a rough copy of the API could be quickly produced, because it is such a critical layer of abstraction we would like to devote a large portion of the fall 2008 semester to API development. Our projected timeframe for API development is to be finished by December 1, 2008 giving us a full two months for development. The remainder of the fall semester will be devoted to revising the prototype code to work on top of the API. We will have our first fully running application based on the Proteus API by the end of the fall 2008 semester. The final part of the phase two, applications research, will take the entirety of the spring 2009 semester. Here we will spend time working closely with faculty and students to develop some benchmark teaching applications as well as novel applications of this interface to manipulate existing software sets.

Project Team Qualifications

Eric Baer and Daniel Delany, third-year Computer Science undergraduates, will both be equally responsible for all aspects of hardware and software development during this project. The collective knowledge of both students includes experience programming in various conventional programming languages such as C/C++, Java, HTML, CSS and processing, as well as expertise of the Max/MSP programming environment. Each student has experience with the construction of computer hardware systems, and an understanding of low level systems through multiple courses in the Computer Science curriculum. Both students also have familiarity with various computer music production tools as well as basic recording and production hardware. Most importantly both students are passionate about this project.

Faculty support will be provided in two principal areas. Dr. Dan Connors in the ECE department and Dr. Michael Eisenberg in the Computer Science department will provide assistance with the technical challenges of the project, including hardware construction and basic data management issues. Dr. John Drumheller, director of CU’s Music Technology program, will supervise and aid Max/MSP coding, use of computer music production tools, and insight into what functionality could be useful to a performing electronic musician.

Other Support

Currently, we have been promised limited financial support (approximately $600) by Professor Dan Connors for the development of a low-cost, working proof-of-concept built with bare-bones materials. While this prototype will help jump start software development and inspire ideas for the next iteration, we are requesting funds for the construction of a separate system which will be tough enough for classroom and installation use. For this reason, contributions promised by Professor Connors do not appear on the itemized budget sheet.

We have not currently secured any additional funding for the phase 2 budget, and plan on funding some portions of the prototype device through personal contributions. For this reason, we have made every effort to search for the most cost-effective parts available and ensure no funds are wasted. We believe this device can be built and implemented inexpensively, using easy-to-find parts and equipment. However, since the project may require additional iterations and refinements in the future, additional grants may be applied for in the coming year, including additional EEF funding, as well as UROP and/or Boettcher Enrichment Grant funds.

In addition to the faculty support detailed above, we have been granted the use of Dr. Michael Eisenberg’s laboratory in the Discovery Learning Center and all tools and equipment (rapid prototyping machine, laser cutter) therein.

BUDGET SUMMARY:

Total Project Budget $	$3969.00
EEF Request $	$3969.00

Outside funding:

Source	*Confirmed? [Y/N]*	Total Amount
Department Contribution	NA	$
College Contribution	NA	$

BUDGET BREAKDOWN:

Equipment and Materials:

Item Name / Description	Unit Price	Quantity	Total Amount
Optics:			
Fire-I board XGAc color camera	$749.00	1	$749.00
2.1mm Wide Lens (No IR coating)	$25.00	1	$25.00
C-Mount IR band-pass Filter	$69.00	1	$69.00
Near IR irradiation	$31.00	1	$31.00
Computers and Software:			
PC	$500.00	1	$500.00
Cycling '74 Max/MSP/Jitter License	$450.00	1	$450.00
Audio/Video Equipment			
PreSonus Firebox Audio Interface	$299.00	1	$299.00
General Audio Cables and Jacks	$75.00	1	$75.00
M-Audio Pro 3 Powered Speaker	$99.00	1	$99.00
Casio XJ-S35 Short Throw Projector	$1,099.00	1	$1,099.00
Radeon X1300PRO Video Card	$77.00	1	$77.00
Building Materials:			
4' x 8' ½" Plexiglas Sheet	$320.00	1	$320.00
Industrial Total locking Swivel Caster	$14.00	4	$56.00
Approximate Lumbar Costs	$80.00	1	$80.00
Building Supplies: Paint, screws, Mounting hardware, etc.	$40.00	1	$40.00
		Total $	$3969.00