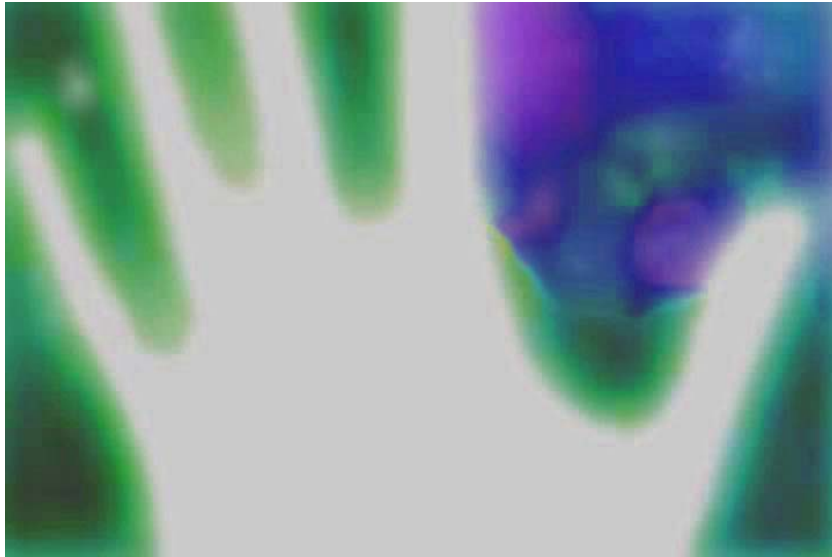


Luminance

An Interactive Art Experience



CSU East Bay Multimedia Graduate Program

5.31.2005

shortattentionspan



J.J. Toothman

Lee Marrs

Scott Wong

<http://sasweb.csuhayward.edu>

Executive Summary

Luminance is a live, playful, interactive art installation in which participants use their own physical body movements to create and influence digital content on a 10' X 7.5' screen. Participants are completely untethered. Their gestures, dancing, leaping and stillness can create image elements on the screen that move with the user and/or move digital content already on the screen.

Luminance is the graduate thesis project developed by Short Attention Span. Short Attention Span is composed of Lee Marrs, J.J. Toothman, and Scott Wong. Luminance was developed to answer the thesis research question, “Can a new methodology be developed to create complex, playful content for multimedia installations whose interactivity depends solely on physical body movements?”

The Luminance installation is developed upon a technical foundation powered by Eyes Web, Floc, and Macromedia Flash. A video camera is connected to a system running the Eyes Web application. Eyes Web is able to take the video signal, process the motion and contrast of a participant’s figure against a screen, and identify a number of data coordinates which provide indicators as the user’s position within the Luminance interactive playground. This data, encoded in a protocol named Open Sound Control (OSC) is transmitted real-time through a Java gateway called Floc. This Java gateway is able to translate the OSC data into XML (extensible markup language) encoded data.

Digital content produced in Macromedia Flash and compiled to run in the Macromedia Flash player. The Flash player is programmed to connect to the Java gateway and receive the XML data stream that originates in Eyes Web. The data is then used by the Flash player to interact and influence digital content projected onto the installation screen.

This technical foundation provided the team an opportunity to investigate the “Three I’s” – Illusion, Interactivity, and Imagination. The illusion of content combined with the interactive nature of content, provided a multimedia experience which provoked the imagination of a participant in Luminance. The content of Luminance is not limited to the sounds and visuals involved in Luminance. Also critical to the content is the interactivity required of the user.

Repeated user testing sessions and numerous cycles of iterative development and adjustments revealed some guidelines to follow when traveling the course of multimedia installation development. Among these is the need to follow a single premise to create a consistent theme. Furthermore, it is critical to “resist living in your head or in your code.” The artistic nature of multimedia installation demands experimentation, testing, and risk-taking.

The Luminance project has many future implications for human-centered computing and the continuing evolution of interface between man and machine. The wireless world continues to expand and eliminate input devices. Devices such as the mouse and the keyboard will ultimately be replaced by, what Marshall McLuhan referred to as, the “Natural Extensions of Man.” Artistic experimentation today can lead to practical implementation tomorrow.

Beyond practical considerations, the possibilities for deeper interactive art creations, that is, participatory art, are now feasible. Participants can expand their own involvement in creative endeavors in an immediate, live experience. It’s a new playground.

Table of Contents

What is Luminance?.....	4
How Does Luminance Work?.....	4
The Research Question.....	4
Conceptual Explanation.....	4
Our Concept.....	5
Interactive Art Methodology	6
The Physical Installation	9
Social Justification	10
The Content.....	12
A Playground	14
Complex Playful Content	14
Technical Details.....	18
Technical Scheme	19
Development Innovations.....	20
Development History and Timeline	22
Team's Personal Assessments.....	44
Appendix A: Influences and Inspirations	47
Appendix B: Selected Resources	51
Appendix C: Budget.....	54
Appendix D: Final Storyboards	55
Appendix E: Supportin Technology.....	58
Appendix F: Misc. Technical Details.....	61
Appendix G: User Testing Forms	66
Appendix H: Usert Testing Consent Form.....	68

What is Luminance?

Luminance is an interactive art experience based upon a multimedia installation that allows participants to be part of the piece. Participants in Luminance use their own physical body movements to influence digital content on a large screen.

How Does Luminance Work?

Participants in Luminance are completely untethered. The user does not use an input device such as a mouse, keyboard, virtual reality goggles, or any other augmented device. Within the installation space, the user's image is captured by a video camera. A computer analyzes the user's image and converts their postures to a set of point coordinates. These coordinates are transmitted to a second computer projecting digital content onto the back of a screen. The point coordinate data is used by the second computer to manipulate the projected graphics and sounds. The user plays an integral part in this feedback loop, responding physically to the projections. In some scenes, this movement changes the content and in other scenes it actually creates the content.

The Research Question

Can a new methodology be developed to create complex, playful content for multimedia installations whose interactivity depends solely on physical body movements?

Conceptual Explanation

The stated research question wasn't the SAS team's first query. Our first question was "What project would be fun to do?" What would be challenging and enjoyable to make that users would find challenging and enjoyable to do? The team was drawn to simplicity and playfulness. We shared what we appreciated doing personally, what multimedia gave us delight, what startling new activities had engaged us. The team began the search among enthusiasms and resources.

We gravitated to two areas of interest that speak to a direct level of enjoyable human activity: sheer physical movement and the ability to transform things in the environment. Many of the most enjoyable of human activities involve body movement – dance, sports, mud wrestling, and charades. And people love to transform things – sand castle building, landscaping, finger painting,

No One Trick Pony

Most current installations in this category are either performances by an artist, exhibiting interactivity that is observed by an audience, or they are constructions whereby participants can have a restricted affect on an element. But, however ingenious and entertaining these latter art pieces are, they all manifest content that is one effect. Very few have compelling content that engages the user for more than a few moments, or attempt even the level of absorption found in a haiku poem.

This is not to sneer at the technical achievement of such work, but the emphasis on delivery systems has too often been at the expense of the content. To create an art piece where the interactivity actually works correctly has been the measure of success. Where Luminance engages users is more in the realm of imagination that's found in poetry or dance.

Our Concept

Our creation process has been designed to explore the interplay of direct physical action, manipulation of images and the user's imagination with iterative user feedback to develop more varied content than is most often encountered.

Interactive Art Methodology

To our research question, the answer is yes. From our work on Luminance, we have developed a set of guidelines that can provide any creator with a process designed to expand the effectiveness of his/her interactive art. This is the recommended methodology for an effective interactive multimedia art installation, no matter its technology, degree/type of interactivity or subject matter:

1.) Brainstorm widely.

Whatever your original idea might be, spend time imagining the wildest and most interesting possibilities of other concepts and variations. No suggestion is too strange or ambitious. Now is the time to explore a full range of “what if.” Keep note of these ramblings, as some of them may aid your installation as it develops.

2.) Thoroughly check out recent creations.

The research should venture far in the field to not only make sure that someone hasn't already done your project, but to locate the support, antecedents and free aid that is now available in many areas. You are not alone. Multimedia art is a mature enough arena to provide supportive communities with free code, references, and valuable advice.

3.) Stay loose, meander, explore.

Human nature craves stability and certainty, even in the midst of some daring creative effort. This is especially true for media, which is predominantly geared toward a stated – often commercial – result. Exactly where a production is headed is a fixed point in the future. Success is measured by how well that declared goal is achieved.

The opposite principle must prevail for innovative art. Producing art is messy, often instinctual, and may remain largely unclear until near the end of production. A freeform pattern is more difficult in a technological zone, but Flash designer/developer Brendan Dawes' term “sketching with code” conveys the attitude and practice where random musings can be pursued in programming as well as in personal journals.

Here is where the developed body of code and other installations' details can serve as a grounding to keep your options open and allow exploration within reasonable time frames for human endeavor. Every wheel need not be re-invented, enabling you to try out, cherrypick, and/or discard several avenues of creation. Or, these meanderings may solidify the notion that you're in very new territory.

However, this means that more of the production process will remain volatile longer than is found in traditional commercial efforts. And you won't know exactly where you'll end up. This will keep many project aspects unfinalized and major alterations may be necessary late in the process. So all project personnel must be prepared for this and willing to pace their energies to accommodate 11th hour marathons if there is a deadline involved.

4.) Discover your main premise and use it as your guide point.

This keystone of good storytelling is a must in order to keep the project on track while exploring the creation of the work freely. The project's premise lies at the overarching level of your intention: your installation is a mysterious evocation of childhood nightmares, or a startling revelation at the juncture of human discomfort and sensual pleasure, or a soothing escape to a fantasy universe.

The premise is your anchor in the heavy seas of exploration. As all the snafus, dead ends, and new avenues pop up during production, whatever decisions to be made are based on the premise. Does this enhance my main point? Is this a detour? Is this so powerful a technique that it'll overshadow my premise? Is it worth it?

5.) Use One Stop Tracking.

Effective scheduling, clear task assignments and a good visceral sense of where the project is progressing can be best achieved by using one production tracking document. This document should include images of the visuals (storyboard panels) as well as the user actions, and creator/team tasks in priority order. By consolidating all these production elements, everyone can track progress, see logjams, and be clear about what needs to be done. This also reinforces a sense of the flow of what users will experience.

If a deadline is involved – “the installation must appear at the Memorial Day BBQ” – then a calendar can be linked to the main document. Games productions have found this

integrated configuration useful for planning long range and short time priorities, especially those concerned with interdependent assignments.

6.) Don't live in your head or in your code.

As soon as possible, create the very roughest prototype of your entire project – visuals and sounds if that's it – to establish another level of inspiration and clarity. Get the installation out of your head and into reality, even if it's only virtual. Seeing and/or hearing the work-in-progress early on can be a good source of happy accidents, wherein even mistakes can be found to be valuable assets. The volume of placeholders doesn't matter.

There are always aspects of a project that are more interesting and pleasurable to work on. A prototype illuminates the relative rhythm in the overall skeleton of your work, pointing up the truly important aspects to be addressed.

7.) Test the real world variables.

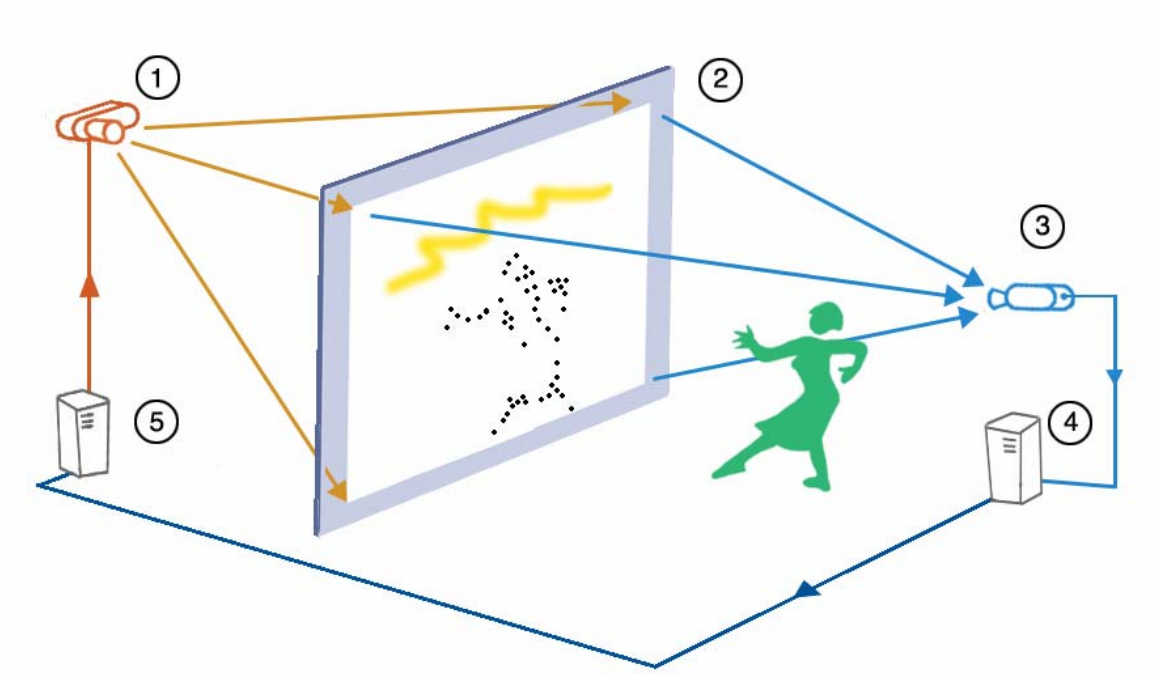
Even if your plan is to express your own apocalyptic vision of string theory, “interactive” designates the user's actions as important. You should have uninformed users play with your creation early in the process (see# 6) and a few times thereafter. Record them in order to combat your unavoidable personal bias.

This is not merely an exercise to evaluate comprehension and layout. This is an opportunity to explore the physical variables of your installation. Its scale, the positioning of physical elements, ambient light, sound system volume and placement, can all reinforce or scuttle your project. Each session should be an adjustment of the overall effect. Would your installation be best experienced in a neon bar, a museum, or a highway underpass?

The most valuable contribution of user testing is to perceive what the users bring to the experience. What stories are they telling themselves about what's happening as they interact? What do they feel is going on? It can be that their interpretation of the installation is more compelling than what you intend. If that's so, what changes can you make to reinforce that new slant while staying true to your premise?

8.) Trust your stream of consciousness and the users' imagination.

It's very easy to fall into a production mode of thinking, no matter the background of a multimedia artist. The spaces between artistic self-expression and user-centered design are a slippery slope. By staying open and receptive to the creation process while observing the "content" generated by the testing user, your art installations can be truly complex and evocative.

The Physical Installation

Luminance in final configuration: 1. Projector 2. Rear Projection Screen – 10' X 7.5' 3. Video Camera 4. Personal Computer networked to 5. Personal Computer

Social Justification

For over forty years, computation has centered about machines, not people. We have catered to expensive computers, pampering them in air-conditioned rooms or carrying them around with us. Purporting to serve us, they have actually forced us to serve them. They have been difficult to use. They have required us to interact with them on their terms, speaking their languages and manipulating their keyboards or mice. They have not been aware of our needs or even of whether we were in the room with them. Virtual reality only makes matters worse: with it, we do not simply serve computers, but also live in a reality they create.

In the future, computation will be human-centered and pervasive. It will be freely available everywhere, like batteries and power sockets, or oxygen in the air we breathe. It will enter the human world, handling our goals and needs and helping us to do more while doing less. We will not need to carry our own devices around with us. Instead, configurable devices embedded in us and/or the environment will bring computation to us, whenever we need it and wherever we might be. As we interact with these "anonymous" devices, they should adopt our information personalities. As users of machine, we should not have to type, click, or learn new computer jargon. Instead, we should communicate naturally, using speech and gestures that describe our intent ("send this to Wayne" or "print that picture on the nearest color printer"), and leave it to the computer to carry out our will.

New interaction systems should boost our accessibility to digital content by lowering the barriers which hinder the usage of the system. To understand this interactivity, a focus on interactive methods which create action-response feedback loops with physical movements must be explored. As multimedia content developers, we need to understand how to create systems that create accurate responses and how these responses are understood by the user. It is these responses which begin to craft the possibilities of content creation.

A project of this type has significance in its discovery of process and method for content creation. Success of multimedia art installations are often dependent upon the environment in which they are encountered and the audience they are reaching. This

project explores content itself and how different types of interactive content create a compelling installation and experience.

With the growth of new means of creative expression, these processes of discovery are necessary in order to determine the potential of the medium. Only through concentration on content can the breadth of the interactive experience be expanded. The processes Short Attention Span discovered should not be considered different than the processes uncovered by painters faced with a blank canvas or a sculptor with large chunk of clay. This journey of discovery served as a roadmap for content creation.

The Content

The intent of Luminance has been to break the barrier of other interactive art installations, in which – however beautiful or engaging – only one particular illusion is created. An extensive examination of previous multimedia installations (in which image recognition was a key component) revealed that their only aim in content development has been to provide a visual or aural response to user movements. Often, the participant in the installation has solely been a trigger, acting to activate a motion-graphics piece. It seemed that a more expanded range of content could be created so that a user could experience a variety of playful delights.

Our main premise was that the user should be able to interact with the installation's content in a simple, direct way, evoking the mood of physical enjoyment associated with perhaps dancing or childhood play. The content needed to evoke this connection, to contribute to a feeling of involvement – to be instantly, obviously, responsive. The user needed to feel that there was a “there” in the screen, it was a place or an entity which “wanted to play” too.

Making It Up

Luminance's content development has proceeded along a searching path much closer to the fine art process than that of traditional product design. Rather than the team envisioning a story or particular “look”, the team embarked on an call-and-response exploration of what the EyesWeb technology could deliver/engender and then what illusion in graphics, animation or video + ActionScript could be created from that. Adjustments were then made to have that illusion be clear enough, strong enough for the users, and so on in a creative loop.

It's Alive!

A dramatic initial realization was that the point coordinates from a user's body movements could itself be visualized on the screen! So those abstract images of the user could be the content itself – the users could “appear” as themselves. Immediately, users

gave the simple images identities and characteristics, largely based on the images' lively behavior.

This discovery greatly expanded the artistic possibilities of Luminance. Not only could participants manipulate content, they could create content – in real time.

Finding the Range

At every juncture, we have tested the possible edges of motion and imagery to see how far we could extend the range of visual and audio content, considering the strengths and restrictions of both the technological and physical configurations. Our changing choices of still and moving pictures has, in turn, caused alterations to the programming of the interactivity and/or the image delivery itself in a constant loop of adjustments, deletions and replacements.

One of the strongest factors in both content choices and delivery has been the physical scale of the installation. Illusions of imagery, depth of field, and motion change dramatically when shifted from a computer screen to the size of a wall.

A driving factor in these reiterations has been the observation of users at play and the users' evaluations of their experience. We enlarged the size of our screen greatly based on observing participants in action. What illusion did this image, presented in this manner, moving at this rate, engender so that the user believed he/she was doing what? By changing which aspect is the illusion strengthened? Quite a few times we were surprised by the users' perceptions and declarations. For example, one data flow configuration on the screen was instantly seen as "bubbles" and another as "tinkertoys". "Hey, fine," we said, and started creating images that would aid those illusions.

Over the Edge

Sometimes we couldn't get there from here. After a month of devoted work, we learned that an animated roadway scene would not work with what now constituted Luminance's range of content. It was too prosaic to establish the necessary illusion for a full interactive experience, no matter how it was redesigned. We found that our initial intent to include these more obvious narrative aspects threw sections of the Luminance experience into a video game zone that transgressed the immediate, live illusion that users had been

experiencing. Participants were already creating their own stories about what they were doing. But this is the pattern in art: finding the parameters by doing.

A Playground

The content of Luminance has three components: moving and still imagery, interactivity, and the resulting imaginary experience – the illusion. The content is a playground of activities, with some scenes appealing more to some users than others. Users immediately felt that they were making everything happen by their movements and they varied in how long they were interested in each scene, so we needed to put them in control of moving on. The participants move from one scene to another by pressing a footpedal in front of the screen.

Complex Playful Content

Nature of Play

Our aim has been to tap into users' natural impulses so that the participants may exercise their bodies and their imaginations. Investigations into the nature of play directly reinforced our initial intentions: to provide an opportunity for voluntary physical activity that can evoke feelings – of amusement, enjoyment, pleasure, surprise. But we became especially alert to the dual purpose often ascribed to some play forms, that experiencing an activity in relaxed ease can engender more complex realizations.

From our user testing sessions, the role of tension is of interest: the expectation that comes from being called to a situation where alertness is needed to participate. Not knowing exactly what happens next automatically quickens users' interest.

Emotional Design

A growing area of product design theory seems germane to our investigations. Don Norman, author of [The Design of Everyday Things](#), has developed a way of categorizing

design by how people process aspects of products. In his new book, Emotional Design, Norman describes three levels of perception in ways that parallel interactive art concerns.

Norman's levels of engagement are:

- **Visceral** : Biologically set preferences, emotional, unreasoning, attracted to bright colors, warm & lit places, and appealing smells.
- **Behavioral** : Ease of use and performance based on practicalities; how well something functions for whatever it does.
- **Reflective** : Messages, culture-based, analytical, relating to self and/or self image.

An effective, playful activity would have visceral appeal, engage the user on a behavioral level of mental processing (be easy to operate), and should have evocative elements to be reflective. His emphasis is on moving beyond functionality to playfulness and fun, and speaks to Luminance's aims in creating engaging content.

"These levels where hope and fear, and satisfaction and anger reside. Deliver on positive expectations and people experience pleasure. Deliver something different than expected, but equally satisfying, and people have fun."

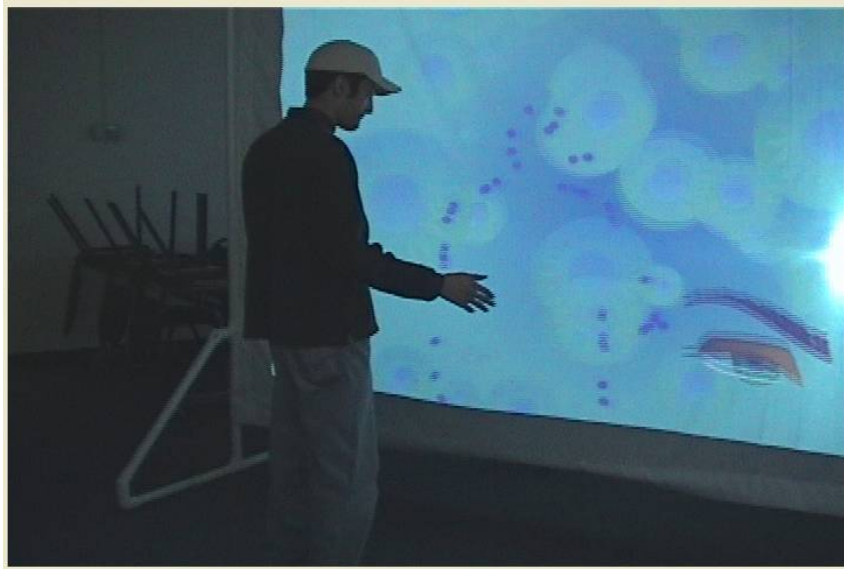
For Luminance

Our installation embodies these considerations to a remarkable degree.

- **Visceral** : Our choice of large body movements as the interactive basis, the large scale of the installation, the pace of moving images, and particular imagery.
- **Behavioral** : our achievements on latency and feedback impressions.
- **Reflective** : our recognition of the users' imaginative responses and the adjustments made in the imagery.

Luminance content consists of a playground of scenes:

The Eye



In this video frame, the participant plays with an animated graphic eye, using the real time dots created by his/her body movements. **TRANSITION:** The eye then disappears in a burst of dots to enable the participant to dance with the dots.

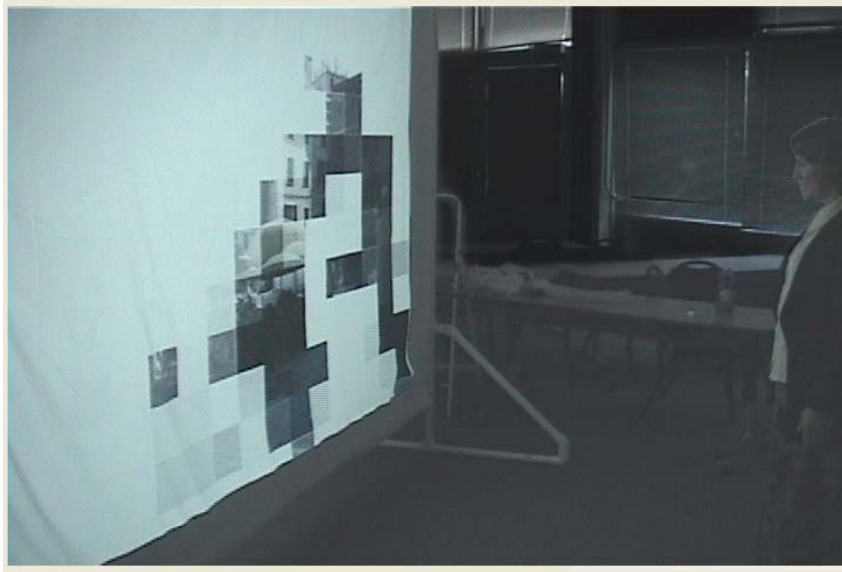
Tinkertoys



The participant's movements are now seen with lines connecting the screen dots, giving a 3D sculptural effect. Shown here at left is the EyesWeb monitor with shadow outline and point coordinate nodes (orange dots). Below the screen is the Do-Wah foot pedal for changing scenes. Next change is the disappearance of the screen dots, leaving angular lines that group and break apart.

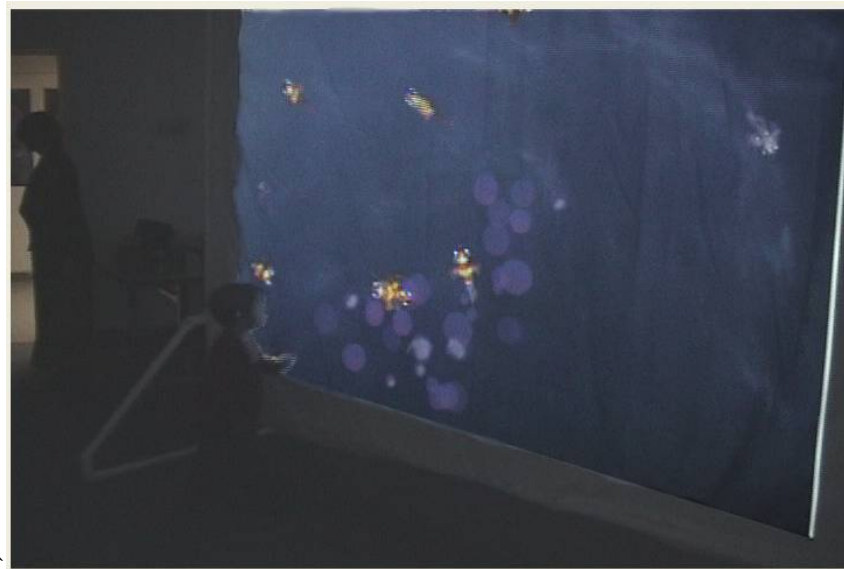
Transition: Increasing large rectangles are generated by the user's movements, blocking out the screen to white.

Foggy Windows



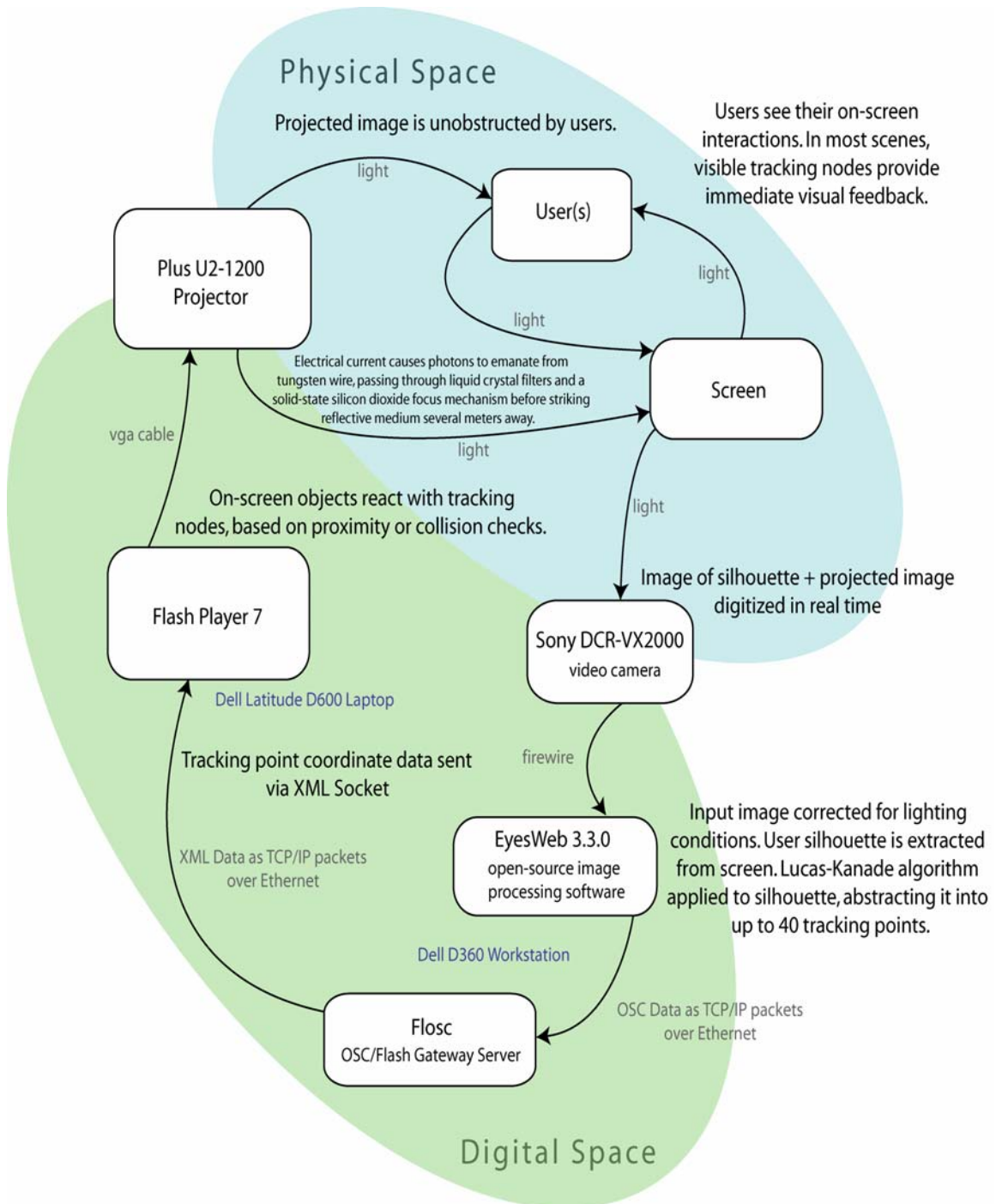
The user's movements "wipe away" the white in rectangular patterns that at first reveal images, then fade back to white in a cascading effect. The last of these images is video.

Fishes



From the video of moving across SF Bay, we transition to under the water, where lively Koi fish interact with user-generated bubbles.

Technical Details



Technical Scheme

- **Video Camera:** Used for capturing the user's image against the video screen
- **Motion Recognition System:** Consists of a PC running EyesWeb and floc. A video camera is connected to this system using Firewire. A custom patch configuration in EyesWeb captures the video input from the camera, isolates the user's silhouette from the screen, and assigns tracking points along the user's outline. The patch formats the XY coordinates of these points as Open Sound Control data packets and transmits them through the floc gateway.
- **FLOSC (Flash Open Sound Control):** floc is a custom java gateway which translates Open Sound Control formatted data (OSC) to XML data. The formatted data is transmitted to a Flash SWF file connected to this gateway over the network.
- **Macromedia Flash:** Receives XML-formatted data from floc. Flash parses the XML and assigns the coordinates to tracking node movie clips. These nodes can interact with other movie clip objects on the Flash stage.
- **Projector:** Projects the output of the Flash client onto the back of the screen, completing the feedback loop.

On the capture end of the system, EyesWeb analyzes the input signal – the screen and the user's shadow. The rest of the screen is filtered based on light intensity; anything brighter than the shadow is ignored. The Lucas-Kanade algorithm picks interesting features around the edge of this silhouette and tags them with “tracking nodes.” The custom patch we wrote in EyesWeb packages these nodes by their name and coordinates and sends them out to the floc server software, running on the same computer. Floc translates this data, the Flash end of our software receives it across the network, and node movie clip objects are moved around on the screen to match what the camera sees.

Luminance does not stream video across the network. The only data sent across the network is coordinate data. The bandwidth requirements are quite low, although Luminance requires low network latency to function effectively. To ensure the lowest latency possible, we run Luminance on its own private network.

Development Innovations

Programming or “Sketching with Code”

This was the development model that we followed. It began with brainstorming about possible interactive possibilities, followed by these general steps:

- (1) Develop interactions locally using mouse inputs. Some example .swfs of this are “Koi Pond” and “Wanderer” (located on the documentation website).
- (2) Convert interaction to respond to Meaningful Movement Data from Eyes-Web test footage.
- (3) Test on live installation setup and adjust parameters & variables
- (4) Code editing and streamlining
- (5) Re-Test on live installation setup
- (6) Color adjustments and other fine tuning prior to full deployment

These steps are necessary due to the physical nature of Luminance as well as the installation scale. The iterative testing cycles on the installation are necessary due to the inability of a development system or sandbox to accurately emulate the variables of the installation, such as CPU processing power, network traffic speed, and lighting.

Content Focus

Instead of continuously addressing the hardware elements and underlying architecture of the installation, the SAS team chose to emphasize the creation of content. We concentrated on content design and behavioral programming instead of system engineering, allowing Luminance to act as a sort of “canvas” for our creative ideas. We remained open to the changes that such a freeform process made necessary.

User Imagination as Part of the Development Process

One of the most interesting discoveries made, as a result of repeated user testing, was the method in which participants created their own narrative and storylines using their own imagination. This aspect became an integral part of the content creation process as the team gravitated towards creating digital content and interaction that would provide an environment which promoted an extensive illusionary experience on the part of the user. In the end, the imagination of the participant became as important to the Luminance experience as the imagery and interactivity.

Development History and Timeline

Fall Quarter 2004

At the beginning, we decided that our best hope for success was to develop an early working prototype of the entire installation. We would have all of the hardware and software in place, as well as a working laboratory prototype of our screen to do testing with. Our goal for the first quarter was to have a working proof of concept, which we met.

September 2004 - Tools Discovery

The first challenge that we faced in meeting this goal was to decide on all of the software and hardware components. We knew that we would need a number of critical components. We needed a software package capable of abstracting an input video stream, a package capable of drawing graphics on the screen based on the input stream, and a means of sending data from one software component to the other. In hardware, we needed a screen, a video camera, a video projector capable of putting out enough light to shine through a semiopaque screen, and computers powerful enough to run the software packages.

The workstations provided to us by the grad program seemed to be capable enough for our needs. The school was also able to provide the video projector and a video camera, though we also briefly considered using a USB webcam. We rejected the webcam as having a resolution too low to work with effectively.

We looked at a number of software packages, without a clear solution in mind as to how to abstract the screen image. We knew that we wanted to capture the user's silhouette from behind the screen; the user's shadow was to be the means of interaction as well as the feedback mechanism. Our two main candidates were Max/MSP and EyesWeb – we gravitated toward EyesWeb since it was free, and didn't look back.

The EyesWeb package came with a fair number of sample “patches,” demos of various techniques and features in the package. We toyed with the idea of extracting position data as color-based blobs, but this approach didn't yield an especially accurate representation of what the user was doing. Eventually we came across their implementation

of the Lucas-Kanade tracking algorithm, which held promise as a method of abstracting edges from a live input video stream.

At the same time, we had decided to use Flash as the software to display our graphics, as 1) we were already licensed for it, 2) we were all familiar with developing content in Flash, and wouldn't have to spend months learning new tools to do the same thing. The problem we faced was in linking Flash to EyesWeb. The EyesWeb user discussion forums suggested a simple, robust path: EyesWeb was capable of outputting data in Open Sound Control (OSC) format. This data stream could be directed at a java server applet called flosc, written by Ben Chun, whose purpose was to serve as a translator between OSC data and Flash's XML socket. This solution proved to be lightweight, robust, and highly reliable.

October 2004 – Configuring Separate Pieces

We successfully connected EyesWeb to the DV Camera for video input. We were also able to create a test EyesWeb patch with a sample .avi file. This test patch would become an integral piece of the ActionScript development process.

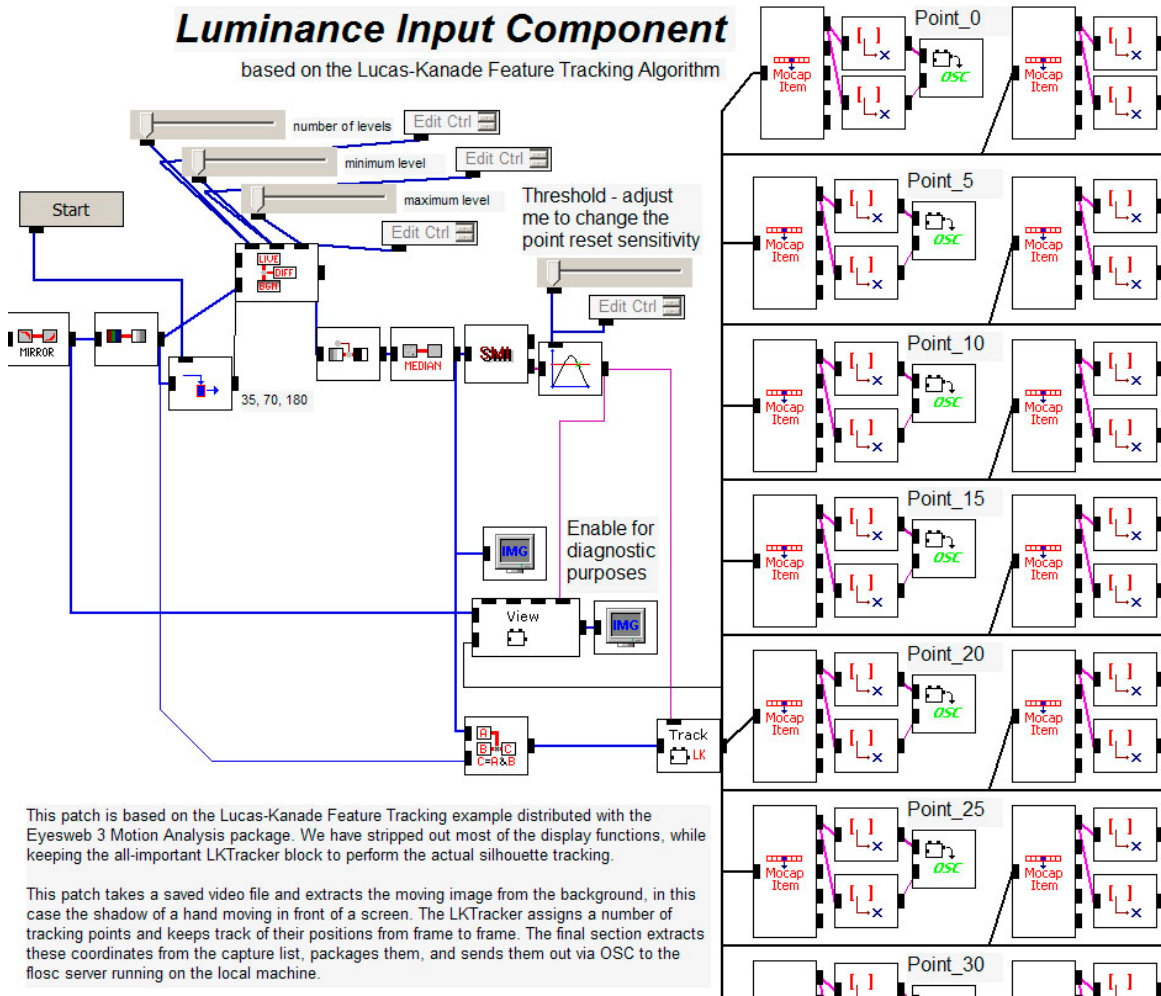
The flosc setup required the installation of the Java Virtual Machine on the host system, to establish connectivity between Flash and flosc.

November 2004 – An opportunity to meet other installation artists.

Lee and J.J. attended opening of Reactive exhibit at RX Gallery including panel discussion with Camille Utterback, Brian Knep, and Scot Snibbe. Observing the live installations and getting an opportunity to discuss with the artists their intents, conclusions and problems greatly informed our subsequent planning.

A flurry of research into the nature of content for art installations, based on new contacts, uncovered no deeper explorations by others, although envy for deeper-pocketed other graduate programs rose. The most complex work seems to be that involving recreations of fine arts techniques, replicating digitally what is done in analog.

November 2004 – Establishing the EyesWeb Capture Algorithm



The Lucas-Kanade algorithm (see Appendix C) as implemented in EyesWeb assigns a number of 'tracking points' to different distinguishing features on an image, and attempts to track their positions across multiple frames. When tracking of a feature fails, for whatever reason, the algorithm attempts to assign that point to a new feature on that image.

In Luminance, the image fed to this algorithm is the shape of the shadow as extracted from the background on the screen, which is based on the lower luminosity in the silhouette regions. If we view the video at this point in the data flow, we see that the source image has been processed into an extracted silhouette; white pixels compose the silhouette,

black pixels compose the background, and there are no features of note inside either of these regions.

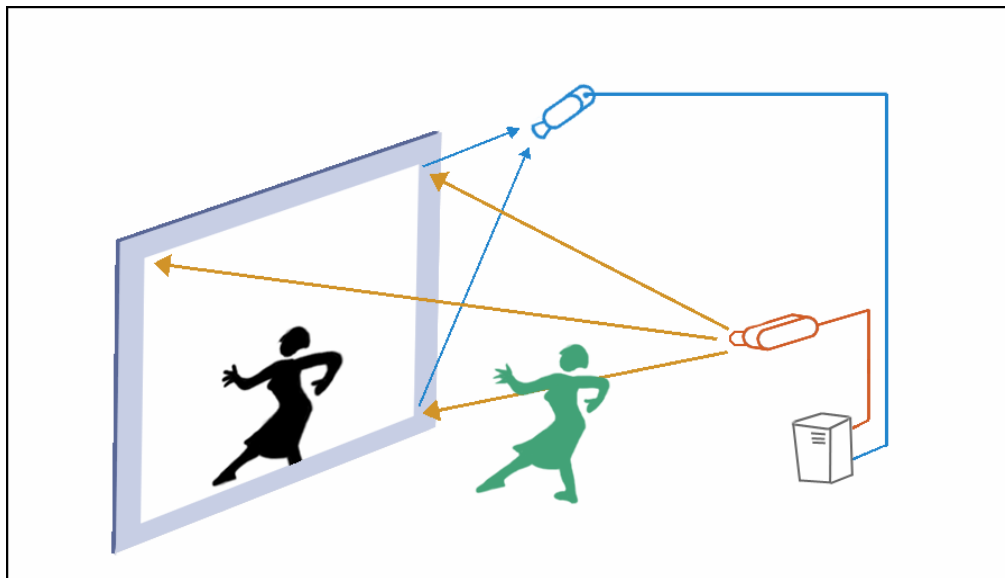
This means that all of the interesting features assigned by the Lucas-Kanade algorithm are clustered along the edges of the shadow, which serves our purpose nicely.

November 2004 - Getting it Running

Our first successful experiment was to render the tracking nodes directly on the screen, proving that we could create a feedback loop in which the user could participate. Our first demo had Flash draw a circle on the screen for each packet of node data, redrawing nodes whenever it got an updated position. A little randomization added to the circle drawing algorithm gave us “bubbles” that formed a rough outline of the user's shadow. These experiments would lead directly to our more advanced components.

Several initial storyboards were created to begin to explore imagery, interaction and themes. Placeholder art and photos were gathered in order to conduct our first testing, choosing a wide range of subject matters and styles.

In late November, we established a full-scale setup in the graduate lab and performed our first user testing. We observed users’ delighted responses and immediate inclination to characterize the natures of dots and bubbles projected onto the screen.



Our first intended installation configuration.

December 2004 – Establishing the Documentation Site

The project web site was launched at <http://sasweb.csuhayward.edu> and blogging by the team became part of the project documentation and discovery process.

December 2004 – The Spectre of the “Hello Kitty” Look

Initial testing revealed a possibility that the settings needed for good shadow position capture might necessitate greyscale parameters for the digital content that would be pastel at best. Tuning and tweaking the capture settings seemed to still indicate that the darkest possible areas be no more than 60%. Heavy consternation from Lee, who has never been a pastel fan and doubted this color range as a source of meaningful graphics.

Experiments ensued to see how well saturation and sneaky hue choices might compensate if this preliminary finding was true.

December 2004 – What Makes a Screen Entity Alive

Lee investigates drifting movement. Whatever transformations are created on screen by the user, there needed to be a constant impression of movement, the illusion of being "alive" that was seen in the random bubbles of the initial prototype tests. Example of drift = slight tap on an untethered balloon. Example of continued pace: one billiard ball hits 3.

This differentiated pattern of movement might create an immediate and clearer feedback to the user as to the consequences of his/her actions WHILE giving the impression that the installation is alive.

Winter Quarter 2005

January 2005 - Investigation of Installation Materials

The materials used for Luminance's physical setup at crucial to both the technical performance of the system, and for the effectiveness of the user's interactions and resulting illusions. Research was done into various sources and possibilities, including information garnered from the iStory Multimedia Graduate Program team.

In February, Short Attention Span held an informative meeting with Otis Brown of Billingsly & Brown Audio/Visual. Mr. Brown provided excellent guidance regarding the types of media projector and video screen that should be utilized in the installation. Whereas we had been concentrating on the screen specifications, Mr. Brown explained the crucial role of various projectors' performance in our achieving data point capture well while projecting colors as vibrant as possible. Specifically, Mr. Brown suggested a projector with a bulb of at least 2000 lumens in order to provide a level of brightness appropriate for video capture. We confirmed that our projector indeed met these criteria, which was lucky, given our overall budget.

Brown A/V generously provided a loaner screen for testing until the end of the quarter. This screen allowed the testing of various installation setups, including rear projection, and also made possible our user testing session.

January 2005 - Test Installation Space

Luminance was installed in a vacant office on the second floor of the Arts and Education building. The test installation was critical in the development of the user experience and interactive content as it allowed for the control of variables affecting the installation, such as the lighting of the space and the color balance of the media projector.

It was also discovered that interactive scripts performed very differently in the actual installation than when being developed with the test EyesWeb clips. This difference was attributed, at the time, to the varying rate of data transfer that occurs in the test installation, due to variables such as a live DV camera feed and network bandwidth issues.

January 2005 – Keeping Movements Simple

Lee explored the use of another variable to affect changes in interactive results – time. That is, having one gesture make a different result when repeated or sustained past some initial amount of time.

Circling the hand would make an object on the screen move in a circle for a while, then (perhaps accompanied by a sound) continuing to circle would make the object smaller. Or make its perspective seem to change by distorting one side of the object to be

smaller. If the object were a rectangle > a door opening.

The aim is to create more complex experiences WITHOUT demanding that the user do more complicated gestures. As in other activities discussed, the pacing of these alterations would be key in their comprehensibility and fun quotient.

January/February 2005 – “Sketching with Code”

Renowned flash designer and developer Brendan Dawes uses the term “sketching with code” to describe a creative process with Macromedia Flash similar to an artist doodling in a sketchbook or a writer carrying a journal. During the winter quarter, we made use of this process to help unlock the creative possibilities of Luminance and begin the development of interactive illusions to help drive the narrative of the installation.

By the beginning of the second quarter, we had a rough idea of the kinds of creative possibilities that we wanted to explore. We decided to tackle a scene that we had dubbed “Foggy Windows,” and met this goal by demonstrating a working scene at the end of the quarter. The user would wipe away parts of a white screen with their shadow, revealing imagery underneath. The drawing API wouldn't allow us to smoothly fade in areas that had been wiped away, so JJ came up with the idea of dividing the screen into a grid of squares whose opacity could be individually controlled.

This problem solved an argument that we had been having about how to resolve the tracking nodes back into an accurate representation of the user's shadow. JJ's idea was to create or find an algorithm that would play “connect the dots,” resolving the set of dots into a rough outline of the user's shadow so that we could treat it as a single object for the purposes of collision detection.

Development on creating such an object proceeded within Flash Action Script programming. During development with pseudocode, it was discovered that the mathematical expressions and logic loops needed to effectively construct such an object were intense enough to slow down the projection system. This would have created a degree of interactive latency between the participant and the content to the point that the overall experience would have been much less rewarding.

Scott insisted that we could ignore the larger outline and treat the dots individually, checking against on-screen objects for range and collision and letting emergent behavior

take care of the rest. We decided that trying to re-create the user's shadow as some sort of vector object was infeasible and tried Scott's laissez-faire method, with some success.

Foggy Windows

The intent of Foggy Windows was to create the illusion of a user being able to “wipe clean” the video screen with physical movements that cast a shadow. In essence, the shadow would be a type of eraser or a rag clearing off a foggy automobile window on chilly morning.

JJ's solution to the previously mentioned drawing API limitations was to develop a grid of squares laid out over a background image, animation, or video clip. Each individual square of the grid has a unique Alpha property that can be assigned or manipulated within the timeline of the piece. The alpha property of an individual squares can be triggered to change when a data point collided with the square. Each square essentially acts as an image mask that can be dynamically adjusted to create unique interactions.

Testing throughout the quarter also revealed that the rate of alpha change on the grid had an impact upon a user's perception of affecting change. Quick changes in an alpha property seemed to indicate higher perceptions of “control” by the user.

Circle Chaos

The Circle Chaos sketch was based upon a sketch first developed during the Fall 2004 quarter in which a dark blue circle of random size, position and translucence was drawn on the Flash stage every second. Our intent was to adapt this sketch for use as a transition within Luminance, so that circles would be drawn at each tracking node data point. With each circle, the radius would increment as while the color faded to white so that ultimately, the circles would be larger than the screen, creating the illusion of clearing the screen with objects that would appear to be getting closer to the participant.

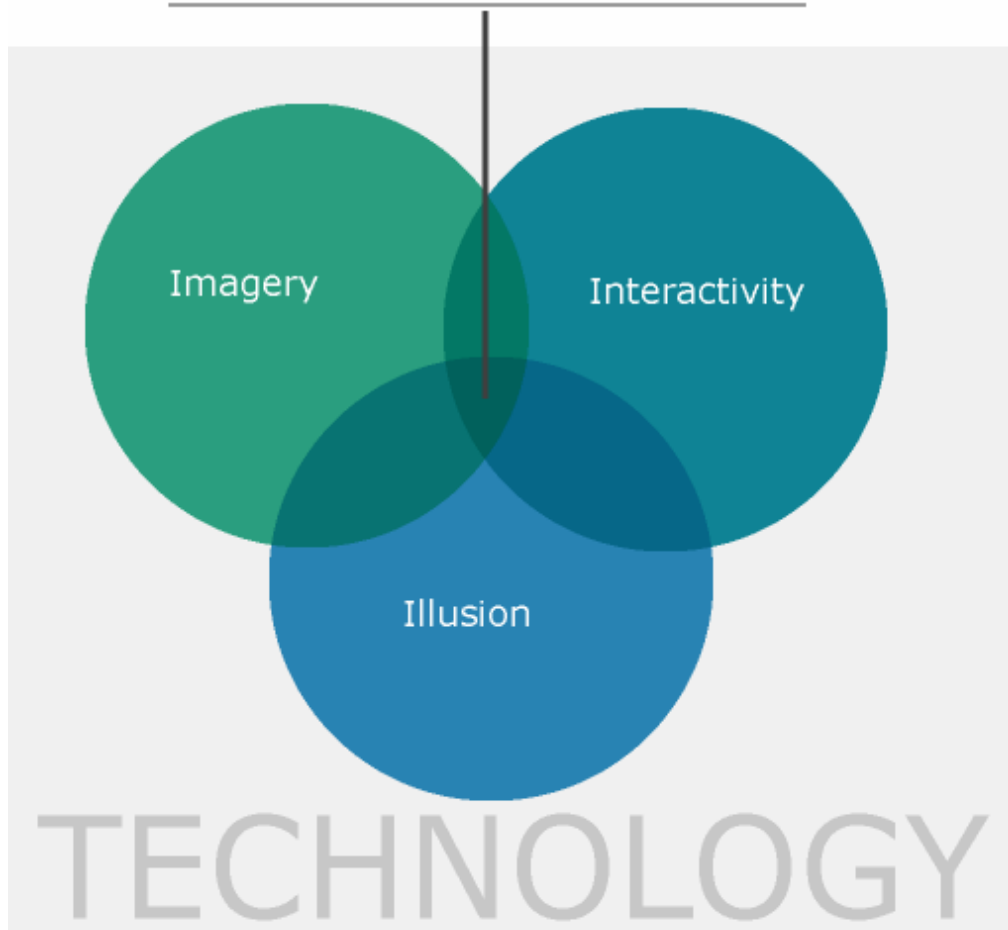
January 2005 - Interactive Content

The second quarter saw a shift in focus of the primary group activity from an investigation and development of technical details to the discovery and investigation of content for the experience. Some of this discovery was based upon production decisions

made for technical reasons, while other decisions were made as interactive limitations revealed themselves to us through ongoing testing.

Successive versions of storyboarded content were created to explore the main aims of the project – direct interaction with ever more complex, playful content – based on the growing number of types of interactivity we were developing, and the technical avenues which were proving unrewarding. In order to engage the widest possible user interest and avoid the well-trod path of prosaic narrative, our content development proceeded with an emphasis on the depth of interactivity as a storytelling arc rather than simple image complexity or traditional narrative. That is, the progression of a user through the content is one of deepening interaction and a more pronounced effect on the virtual space.

The intersection of Imagery, Interactivity, and Illusion is the primary area of exploration in Luminance. Technology serves as the foundation for "The 3 I's."



February 2005 – Initial Content Production Charts

Using the latest versions of the storyboards, production charts were created to delineate the project's content, activity and team tasks to accomplish the installation. The charts encompassed the digital content assets, the content's behavior based on the user's movements, the sound needed and the assignment of tasks connected with all this. The charts accompany weekly to-do lists for all team members. The charts were working documents, revisited and revised via group meetings, email threads, hallway chats, shouting matches, veiled threats, burning effigies, and other means of group discovery and exploration.

February 2005 – Color Balance Results and Off-Screen Roadway Tests

Using a rear projection screen cuts down on the automatic brightening that usually occurs in projection, making the shift from computer display screen imagery to screen less of a risk re data capture. Range of greys can be as dark as 70 – 75%. Drinks all around.

Based on wider graphics range now possible, Lee began planning Roadway scene in Flash with placeholder elements. Initial discussions indicate all movement will need to be in ActionScript rather than using motion paths, etc. Lee launches into deeper code inquiries. So many variables at work at once make initial version mouse-clickable a necessity.

March 2005 – First User Testing Session

On Friday, March 4, Short Attention Span conducted live user testing with a half dozen participants, both strangers and acquaintances of the team. Participants were not given any instruction or guidance other than to stand within the installation space. Content of the installation was periodically changed on the fly to show and test different types of interactive content while Short Attention Span observed their activity. Each user was videotaped. Each participant spent roughly 10 minutes in the installation, and was then asked to complete a short survey.

The testing was most useful to reveal areas in which Luminance excels, while also revealing the areas in which improvement would benefit the overall experience.

This testing revealed several things about Luminance:

1. We encountered a high amount of latency (up to 20 seconds) in everything but the simplest of our scenes. This confused users, who often couldn't tell if they were interacting with the screen.
2. Our tracking nodes seemed to be floating around the screen, taking up strange positions when no one was creating shadows on the screen. This would cause them to trigger components on the screen that they were touching, like squares in Foggy Windows.
3. Much of our imagery was too dark, causing EyesWeb to mistake projected elements for shadows and assign them tracking nodes. Dark screen elements would "self-trigger" in a positive feedback loop.
4. Users couldn't tell what they were doing because their own shadows were in the way. They also had to cope with the glare of the projector on their backs, which was especially difficult for anyone wearing glasses.
5. Regardless of what was actually happening on the screen, users made up their own stories based on what they perceived. They were "wiping off the screen," they were "making the scenes change," etc.

March 2005 – "More Magic"

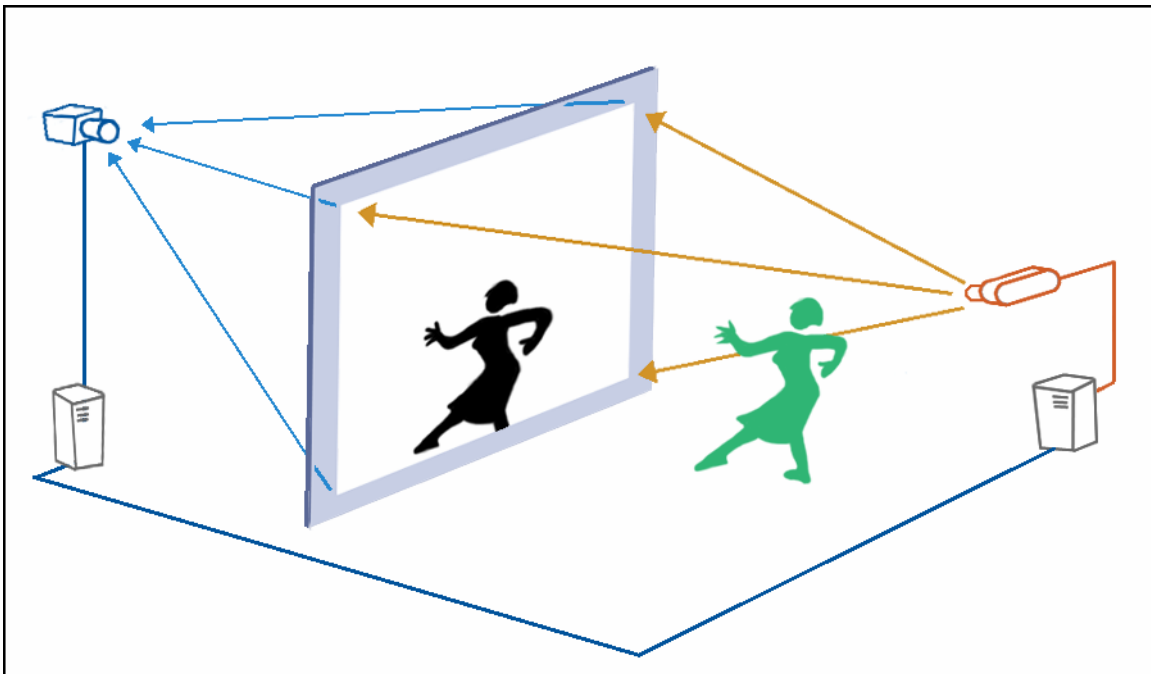
A very late development in the second quarter led us to decide to flip the whole system around, so that the user was standing on the camera side of the screen, while the projector beamed the image at the back of the screen. JJ had argued for this approach from the beginning, with Scott holding the dissenting opinion that the user's shadow was the only clear means of providing feedback.

Scott was eventually swayed by an argument made by Mr. Wayne Chan, another student on the program. To paraphrase, Mr. Chan described the backlit screen as "more magical," and that having actions take place on the screen without something as stark as the user's shadow was "less mechanical."

It took surprisingly few adjustments to have EyesWeb isolate the user's silhouette from the screen. Lee solved the problem of giving users feedback on their own position by having the system draw the tracking nodes on the screen in almost every scene; to the user, they act almost as a cluster of cursors.

This change, however, introduced new complexities to the user interaction. We now have to shepherd users away from the capture camera, as they can totally obstruct the system's view of the screen and render Luminance functionally blind. Users make a larger silhouette as they stand further away from the screen, as before, but it is much less obvious now. The color of a user's clothing now makes a significant difference in how the system perceives them. If they are wearing light colors, the algorithm tends to assign nodes in a dense cluster inside their bodies instead of in an outline around their bodies, meaning that some people really are treated differently.

On the other hand, the user can see what is happening on the screen now.



Our second installation configuration, with the video capture in rear.

March 2005 - Eyes-Web Fine Tuning & Color Adjustment

Through iterative testing, we identified performance issues that prompted adaptation and modification within our EyesWeb patch. We needed to be able to change the filter settings within EyesWeb on the fly, due to the changing lighting conditions from session to session. Understanding the adjustments needed was critical to the development of consistent contrast within the digital content being developed. We also found ways to fine-tune the filter in EyesWeb to accommodate a somewhat wider range of lighting conditions.

Short Attention Span –Lee Marrs, J.J. Toothman, Scott Wong

In the artwork, Lee solved most of the image feedback problems by meticulously going through and tuning the color balance of the artwork to stay above a certain darkness threshold.

As a result, Eyes-Web is effectively “tuned” to work with content whose greyscale range is 70% gray or lighter. That is, at levels darker than 70%, Eyes-Web is unable to distinguish between digital content from the projector and participant shadows being cast upon the screen. The content imagery must lie within this tonal range for the data points to be captured accurately.

March 2005 – Conversion to ActionScript 2.0 & Object Oriented Programming Model

Late in the winter quarter and after numerous testing sessions, we made the drastic decision to convert all Flash code from ActionScript version 1.0 to ActionScript 2.0.

ActionScript 1.0 was originally chosen due to the group’s current intimate knowledge of the programming and environment and the abundance of robust code samples shared by the Flash developer community that have been perfected over the years. However, testing throughout the winter quarter saw a need for a code structure that would allow for faster and more robust control of our on-screen objects. We saw the object-oriented nature and object class development possibilities of ActionScript 2.0 as a solution.

Part of the solution to the latency and the tracking node problems was to convert our architecture to use the object-oriented approach. The behaviors of interactive on-screen elements such as the squares in Foggy Windows and the tracking nodes were re-created as ActionScript 2.0 class files with native behaviors. The tracking nodes received behaviors such as parking off the screen when they were longer receiving new input, solving the “ghosting” problem, while the squares had their fade methods and behaviors integrated. In addition to substantially improving performance, this conversion accelerated the development of new content; once written, new interactive objects could be spawned onto the screen with a simple framework of code to manage them.

The decision to change ActionScript versions came late in the winter quarter and only after exhaustive levels of testing with version 1.0 code did not achieve the desired results. This change impacted the state of content production. While content development

proceeded at a comfortable pace with effective storyboarding, code production and realization of content was delayed due to the decision to change versions. Fully aware of the risks inherent to a conversion of this scale, we made this in preparation for full content production during the spring quarter.

March 2005 – Prototype Demonstration at Forum

A prototype of Luminance was presented to the Multimedia Graduate Program Forum in March 2005. Once again, the amount of space and time to set up was impressive. After graduation, we can all get moving company jobs.

As usual, the installation entertained and intrigued people, but scenes keep stacking up, not clearing out data. The latency actually seemed like a “Me and My Shadow” effect to some.



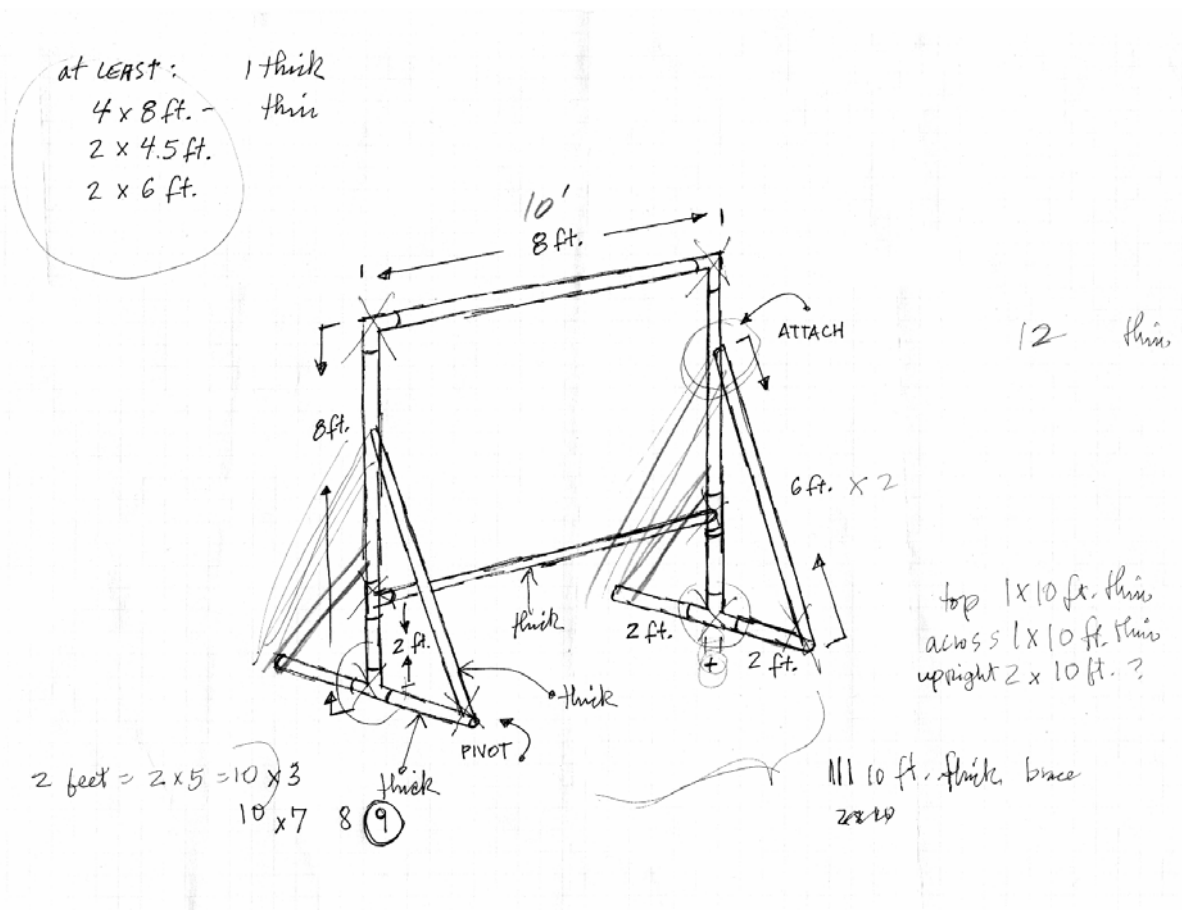
Left to Right: Lee Marris, J.J. Toothman, Scott Wong

Spring Quarter 2005

April 2005 - Installation construction

Having been asked to return the test screen to Otis Brown, the team assembled the installation screen frame out of PVC tubing in April 2005. The frame is self-standing, portable, and assembles without nails, screws, or bolts. After an exhaustive search, Lee managed to find a treated, flame-retardant rayon screen from Rose Brand Theatrical Supplies. This screen was custom made to be part of the final installation and dazzle users with its showbiz panache .

Having assembled the installation parts, the team established residency for Luminance in room AE160.



The design sketch for Luminance's screen.



Contrary to popular belief, we were not building a giant bong.



Measure once, cut twice.



Glorious reign of SAS to last 1,000 years!!!!

April 2005 - Harder Better Faster Stronger

We entered the third quarter with a fleshed-out storyboard and a stable but slow installation setup. On the technical side, our goals were to drastically improve performance and to provide more complex interactions for the users to play with. We met both of these goals with a few major breakthroughs.

When we decided to treat the tracking nodes individually, we realized that we would need some sort of algorithm that could quickly evaluate our nodes for proximity to other on-screen objects. Some searching led us to a proximity management ActionScript 2.0 class written by Grant Skinner.

Objects attached to an instance of this proximity manager would be abstracted to cells on the screen, using a technique almost identical to the one that JJ had come up with for Foggy Windows. Given a reference to an object on the stage, the proximity manager will return references to all of the objects in the neighboring cells. Doing this filters the number of more complicated distance checks between an on-screen object and the tracking

nodes down to a reasonable level; figuring out how to incorporate this technique into Luminance was a significant achievement in improving performance.

Parallel to this, we learned how to push and pull objects based on proximity. One demo, from a set of lecture notes called “Hatching Computational Creatures” published at levitated.net, showed us how to spawn dots on the screen that would randomly wander around, like microorganisms in a drop of water. Another demo in the same set of notes called “Barslund Repulsion” showed us how to have objects squish away from a home position based on proximity to the mouse cursor. Scott combined the algorithms in these demos to make a set of dots that would wander around on the screen, but also smoothly flee the cursor.

The goal of this experimentation, though, was to find a way to create the “fish pond” scene that we had envisioned on our storyboard. The fish had to roam around in a way that would convince the user that they were real fish, they had to be capable of interacting with a set of nodes rather than the single point of a mouse cursor, they had to change swim animations based on how fast they were going, and they had to face the direction they were heading.

Some further experimentation with the wandering dots demos yielded a pulling behavior not unlike gravitational attraction. By manipulating the velocity of the dots, rather than directly manipulating the position as before, we created a very rudimentary physics engine. Attracting and repelling objects alter the trajectory of other objects instead of directly pushing or pulling them, which we felt would be much more natural for the users to grasp. In practice, this means that the user can do things like bat the eye across the screen, catch it and fling it, or carefully coax it to follow their hand.

April 2005 – Animation Deliveries

Along with continued graphics adjustments to photos, Lee began delivering successive versions of our initial animated “character”, the starting eye, the transition “exploding” eye, and its animating background. Based on Flash pseudophysics, it was then ascertained that having code based swfs wouldn’t work well with the overall installation’s programming. So back to motion paths, etc. for the animated elements.

Large scale of screen plays havoc with relative motion speeds of Eyestart moving background parts, resulting in many versions. Animated fish cycles are a workable winner from the getgo as basic elements that Scott then fit into the larger scheme of interactivity. We go through many alterations of scale (ha!) and volume of fish at which size.

April 2005 - Sound Design

Sound was always considered to be an important element to Luminance. The variety of scenes and the evocative atmosphere the visuals provided were improved with the addition of some short sound loops.

J.J. designed and constructed some compositions with the use of royalty-free music loop DVDs. The compositions were composed with conga sounds that added to the playful nature of the installation. Bowl turning and bell sounds were used for the mysterious and investigative scenes of foggy windows. User testing showed that the sound additions were well received and that additional sounds could be used to indicate a scene change or event trigger. Thus, J.J. composed a larger drum kick sound to be used when a participant activated a scene change.

April 2005 Farewell Roadway

After serious work and a complex result, the Roadway scene just won't fit with the other sequences and seems like it's from some other project – however well it relates to the Eye look. Moving this far toward explicit content has defined the character and scope of the other content, though. We now clearly know what current Luminance content is and isn't.

May 2005 – Video Shoots

Testing had shown that participants in Luminance enjoyed seeing moving video and animation in addition to seeing static photos and graphics. The project team desired to put some video footage in a scene for foggy windows. So J.J. spent a Friday morning seeking some water scenes and shooting some video in Golden Gate Park. The videos shot at the Botanical Garden and the Japanese Tea Garden were very murky and dark and did not add to the illusion of depth which was desired. So J.J. shot some additional video footage from the ferry to Tiburon. Provided with a bright sunny day, footage of some clear shallow

water that was slowly moving provided to be an effective background to the fishes scene. Though some color adjustments were needed to increase the brightness of the midtones. Additional footage shot from the ferry showed the bay water in the foreground with the San Francisco landscape in the background. This footage was chosen to be included in the foggy windows scene.

May 2005 Speed++

The last major technical hurdle to overcome involved the performance of Luminance. Through our last user testing session, we were coping with latency in the input of up to ten or even fifteen seconds; the user's actions were literally being delayed so much that they had trouble distinguishing their actions from random things happening on the screen. We had two clues to work off of:

First, JJ and Scott had been working on different scenes, isolated from the larger Luminance structure, on PCs running Windows XP and EyesWeb. A test video file substituted for a camera input, the floc server to bridge the data to Flash was run locally, and the network traffic was looped back to the Flash client running on the same machine. While not quite powerful enough to smoothly run everything, Scott and JJ both noticed that the video input and the output in Flash were always within a quarter of a second of one another, nowhere near the latency of the production version.

Second, the production version of Luminance had the Flash client run on Lee's G5. With EyesWeb running the test signal on our Dell D360 workstation, and the G5 receiving the data stream across the network, we observed severe latency. Traces across the network reported a small amount of latency inherent to the system, but nothing to the degree that we were witnessing. Utilization meters on the G5, though, indicated 100% processor usage.

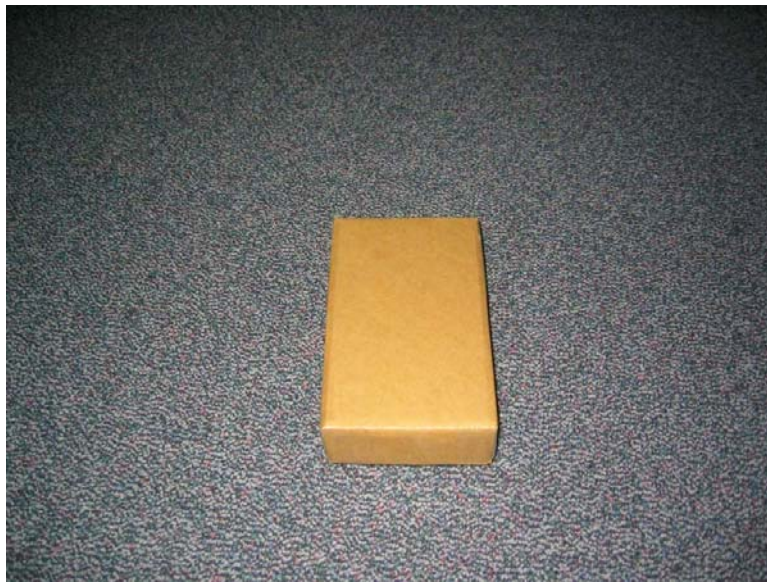
We set up a simple experiment. The latest version of Luminance at the time was compiled into a single stand-alone swf file. This file was distributed to both the G5 and to another D360 workstation in the lab. EyesWeb and floc were started on our server, with EyesWeb configured to display the test signal on the screen so that we could compare the position of the playback with the delayed nodes in Flash. With the G5 connected, we saw up to ten seconds of lag between the playback and the on-screen interaction. With the D360

connected, running the same swf file, to the same server, across the same network, there was less than 250 milliseconds of apparent lag.

A Dell D600 laptop was provided for us to run Luminance's display portion, which was quite adequate for our needs. We had to do some slight color calibration on our projector, but the swap was otherwise uneventful, and solved our latency problems.

May 2005 – User Testing

Final, festive round of users trooped in to play. Much richer experience than before, good feedback again. Users continue to create what they feel they are experiencing. The changing from scene to scene is too abrupt, users are taken out of the moment whenever it happens, even if they've become visibly bored with that particular scene.



Sophisticated interface device – Luminance's Do-Wah Pedal.

May 2005 - The Do-Wah Pedal

Late in development we had to deal with the problem of changing from scene to scene. Our initial storyboards had called for the user to initiate changes based on some screen-based interaction. Some scenes were to transition based on being interacted with for a set period of time, while some others were to require the users to manipulate on-screen elements to trigger a transition. After our second user testing session, as well as a good

amount of internal testing, we found that the users generally couldn't control on-screen interactions precisely enough to for users to accomplish specific goals.

We decided on two simple methods for transitioning scenes. The first would be a simple timer, easily coded in ActionScript. The second would be some sort of button that the user could press to control their own advancement through the scenes. We briefly considered having some on-screen hotspot that they could touch with their silhouette, but rejected the idea of having something static that would interfere with the artwork. We considered having a button or mouse hanging from the frame of the screen that they could click, and this idea evolved to a foot pedal.

A foot pedal was consistent with our goal of having untethered, hands-free interaction. JJ added code to Luminance that would advance from one scene to the next based on a mouse click, and purchased a presentation mouse that would operate at a greater range than conventional wireless mice. Scott constructed a housing for the mouse that would allow users to click the mouse with their foot. The addition was cheap, worked reliably, and most importantly, allowed us to incorporate duct tape into our project.

May 2005 – The Blue Color Cast

The shift from a Mac computer as the projecting source thankfully did not darken all the digital content so much that major adjustments had to be made. So Lee did not slit her wrists. However, the content did now have a dramatically blue cast to all the imagery and a softening of contrast. Tech wizard Gavin Farrington thought the guilty party was actually the projector. A very very long session with both the projector's color space and the Dell laptop's color space finally improved the situation. But it is still fortunate that so many of our images are of water.

Team's Personal Assessments

J.J. Toothman

J.J.'s growth during the graduate thesis year occurred on many fronts, with the knowledge gained in technical area the simplest to pinpoint and define. Luminance represented the first extensive "off-the-monitor" multimedia production project on which he had worked and also the first opportunity to work with a physical and/or analog computing interface such as Eyes-Web. Eyes Web's use of a Java gateway to communicate with Flash offered the opportunities for him to understand the detailed nature of XML datatype definitions.

The conversion in the Winter Quarter to Actionscript 2.0 was his most involved experience with Object Oriented Programming structures to date with the advantages of class definitions now clear to him. The nature of the Luminance installation and the technical limitations discovered throughout the year forced him to be write programming code that was streamlined and more efficient that that required of closed high-bandwidth networks.

In the design sense, the size of the installation and "off-the-monitor" deployment led to an increased awareness of "animated illusion" and use of depth perceptions (the "z-dimension" as we referred to it). These illusions led to additional knowledge gained in interactivity design.

The most compelling area of growth was gained from observations during user testing session, watching how participants interacted with Luminance. It was revealing to discover the critical nature of the feedback loop in the installation and how that loop, when performing effectively, allowed participants in Luminance to be imaginative with the free play.

On an artistic level, it is easy to see that users are capable of creating their own narratives within the environment. This type of learning can lead to additional development in human-computer interfaces that are currently found on the web and interfaces being developed for the future.

Lee Marrs

Experienced in creating linear, non-linear, and seeded (or dynamic) storytelling, Lee's most dramatic learning curve regarded the power and effectiveness of the user-elicited story as found in the Luminance participant sessions. Previously her fine arts creations had been entirely abstract and her storytelling work very explicit. The strong response by users from what seemed like very vague visualizations was a revelation.

Also the building of such a large construction in the physical world was a source of personal learning. Although many of her previous animated and special effects work had been projected at immense sizes at conventions, she had never witnessed live, interactive behavior on such a scale. Observing responses live is so much more powerful than hearing from readers several months after seeing a graphic novel.

The crucial element in the art installation content seems to be the perceived source of content generation – when the user feels that he/she is creating the imagery, not just controlling it – then the emotional investment ramps up. This brings the concept of seeded storytelling into a much deeper realm of possibilities, whereby seeds of ideas and patterns can evoke complex storytelling on the part of the participant.

This “letting go” of narrative was a painful experience for a veteran Virgo, but a valuable new tool for creative futures.

In the realms of technological learning, this was the first time Lee had to deal with code up close and personal, having been protected by layers of minions in the past. Lee found a breakthrough in understanding of programming in the virtual physics tutorials of Keith Peters, whose patiently step-by-step code revealed an easily visualized link between animation and programming. Lee could grok it. Code ceased to be Castor Oil, although not yet chocolate. So more code-based ActionScript work is in the offing.

In setting up the elaborate-if-ill-fated roadway animation sequence in Flash, Lee gained a deeper experience in the strengths and weaknesses of Flash, clever workarounds and perspective/trajectory tricks that will serve her well in future projects.

Scott Wong

Mr. Wong started this year with little experience in total project development. From concept to delivery, Luminance was the first full project that he has been involved with. Aside from technical growth, he learned about project planning and management, development methods like iterative development and rapid prototyping, and later, methods like alpha testing and user evaluation.

From a technical standpoint, Luminance has been his most ambitious technical undertaking to date. We started Luminance with no prior experience in live motion capture technologies. Scott's accomplishment in this area was not only learning how to use the various platforms necessary to capture meaningful motion data, but also to learn to learn how to bridge that data from one platform to another to create a viable installation.

From a tools standpoint, simply put, Scott learned more advanced programming techniques. He gained a foundation in object-oriented programming, something that he was previously lacking. He learned how to create computational behaviors, and how to manage populations of objects. He learned how to manipulate equations to simulate basic physics. Because of performance constraints imposed by the chosen rendering platform (Flash), Luminance's programming required high attention to efficiency, which improved Scott's programming technique.

Appendix A: Influences and Inspirations

Luminance was informed and inspired by many previous efforts of research, artistic exploration, and experiments. Luminance has been influenced by projects and theories of human computer interaction, human sociology, and learning theories.

Mine Control

“We believe that art can be fun, playful, and simultaneously thought provoking.”

www.mine-control.com

Led by artist/programmer Zack Booth Simpson, Mine Control has produced exhibitions exploring interactions between users, projected screens, and shadows. Beginning with more rudimentary demonstrations of particle behaviors in 2000, the group has since explored such topics as collaborative game environments, visualization interfaces, as well as installations using props and flashlights as the tools of interaction. Most of Mine Control’s exhibits are based on Simpson’s custom-authored “Shadow Garden” platform.

Myron Kruger – “Responsive Environments”

Myron Kruger’s work in the 1970’s investigated the interaction between machine and humans, with the interaction serving both as interface device and as a tool to develop social relationships. Kruger used concepts of responsiveness to create artistic experiences and environments in which the ultimate goal was for “full-body participation in computer events that were so compelling that they would be accepted as real experience.”

Marshall McLuhan stated that the “massage is the medium.” Similarly, Kruger’s work explored the notion that “response is the medium.” The idea of full-body interaction creating action and reaction systems is a core fundamental characteristic of Luminance.

Camille Utterback – External Measures Series

<http://www.camilleutterback.com>

Interactive installation artist Camille Utterback explores interactive mediums that provide a fertile environment to explore the connections between physical bodies and the

myriad of representational systems possible in the digital realm. Utterback attempts to bridge the conceptual and the corporeal in exploring how people use bodies to create abstract symbolic systems, engaging people's bodies instead of just their fingers and eyes. Interactive systems determine the grammar of interaction with digital media, and ultimately its possibility for meaning. Her External Measures series attempts to create aesthetic medium which responds fluidly and intriguingly to physical movement in the exhibit space.



Utterback's influence upon Luminance was to develop an interactive installation that respond to a participant's actual gestures, body language, and physical location with the installation space. By developing physical-digital systems that engage people's bodies instead of just their fingers and eyes, the intent is to refocus attention on the embodied self in an increasingly mediated culture. In addition, Utterback places a great deal of emphasis upon aesthetics.

Norbert Weiner – The Human Use of Human Beings

Norbert Weiner's insights into human-machine interactions are the premise behind all human-computer interactivity and interface design. Weiner observed that the quality of data transmission between man and machine is affected by system feedback and noises. Weiner also promoted the idea that interactions between man and machine should be based upon natural communication methods from person to person.

The influence upon Luminance should be obvious: the communication between man and machine in Luminance depend upon natural physical communication of man.

Bill Viola

The work by multimedia artist Bill Viola influences the Luminance project in a number of ways. The first way he has been an influence is merely by the scale he uses for his video installations and the high quality of production value in his projects. But most importantly, his choices of content are influenced by artistic works of other mediums. For example, his video piece “The Greeting” is inspired by Italian Renaissance paintings.

Jean Piaget

Jean Piaget’s theories of cognitive development influenced Alan Kay’s developments of the graphical user interface and the Dynabook while he was at Xerox PARC. Piaget outlines four stages of cognitive development which had certain characteristics. Short Attention Span was inspired by Piaget's theory postulating that people develop *schemas* (conceptual models) by either assimilating or accommodating new information. These concepts can be explained as fitting information in to existing schemas, and altering existing schemas in order to accommodate new information. This was a cornerstone of the initial content developed within the user interface

Reactrix

<http://www.reactrix.com/>

Reactrix is a company in Menlo Park, California which specializes on interactive marketing installations. They have developed a software/hardware configuration which has been installed in high traffic consumer areas. We spent time observing an installation of Reactrix at the Sony Metreon in San Francisco. Users interacting with the installation, particularly children, became immersed in the digital graphics on the floor and often spent lengthy amounts of time discovering and playing.

Short Attention Span aimed to produce compelling enough content to sustain audience participation for such lengthy periods of time. The playfulness of Reactrix systems and digital content presented an excellent example.

Burning Man

<http://www.burningman.com/>

The annual art festival Burning Man encourages not only creative artistic expressions but also promotes an atmosphere in which everyone is a participant. Similarly, Short Attention developed Luminance as a multimedia installation that demands active participation. Luminance is not for the casual observer.

kirupa.com

<http://www.kirupa.com>

An ActionScript community and tutorial site, from which we learned more advanced scripting techniques.

levitated.net

<http://levitated.net>

The examples and demos generously provided at levitated.net taught us how to create more advanced effects and behaviors with our art elements in Flash.

Appendix B: Selected Resources

Hardcopy References:

Packer, Randall, and Jordan, Ken. Multimedia: From Wagner to Virtual Reality. W.W. Norton & Co. New York. 2001.

Bolles, Edmund Blair. A Second Way of Knowing: The Riddle in Human Perception. Prentice Hall Press. New York 1991.

Bruce, Vicki, and Young, Andy. In the Eye of the Beholder: the Science of Facial Perception. Oxford University Press. Oxford.1998.

Curtis, Hillman. MTIV: Process, Inspiration and Practice for the New Media Designer Pearson Education. 2002

Rock, Irvin. Perception. Scientific American Books, Inc. New York. 1984.

McLuhan, Marshall. Forward Through the Rearview Mirror. Prentice Hall Canada; (1996)

McLuhan, Marshall. Understanding Media. The MIT Press. 1994

Viola, Bill. Going Forth By Day. Distributed Art Publishers; (July 2003)

Meadows, Mark Stephen. Pause and Effect: the art of interactive narrative. New Riders. 2003

Norman, Donald A. Emotional Design: Why we love (or hate) everyday things. Basic Books. 2004
The Design of Everyday Things. Currency Books. 1990

Moock, Colin. Essential ActionScript 2.0. O'Reilly. 2004

Web References:

<http://www.setpixel.com/content/?ID=105>

Firecaster

<http://www.creativityandcognition.com>

Creativity Cognition Studios

<http://www.uni-weimar.de/~bimber/research.php>

Augmented Reality in Real Environments

http://www.artcom.de/index.php?option=com_acprojects&page=6&id=7&Itemid=115&details=1&lang=en

Images that compel folks to do things, close relationship between physical activity & resulting change in images (ice breaks, water ripples) which evokes specific & clear activity such as stomping, shuffling.

<http://interactivity.stanford.edu/projects/barehands.htm>

"BareHands uses recognition of the shape of the touching hand (one finger, two fingers, side, etc.) to combine the convenience of direct touch with the power of multi-key interactions."

<http://www.3dluvr.com/voltaire/plaything>

<http://www.idonline.com/imdr02/body.html>

A camera-based tracking system monitored the coordinates of overlaid shadows in real time, using movement as an agent of change.

<http://www.flatblackfilms.com/>

mesmerizing, translucent, shapes.

<http://www.unfinished.com/index.html>

Recent Discoveries from The Dept. of Shape Research

<http://www.fp.ac.nz/>

Functional Play

<http://hci.stanford.edu/seminar/abstracts/03-04/040526-mankoff.html>

Enhancing User Experiences in Ubiquitous Computing.

<http://oxygen.lcs.mit.edu/Overview.html>

Pervasive, Human-Centered computing

Media:

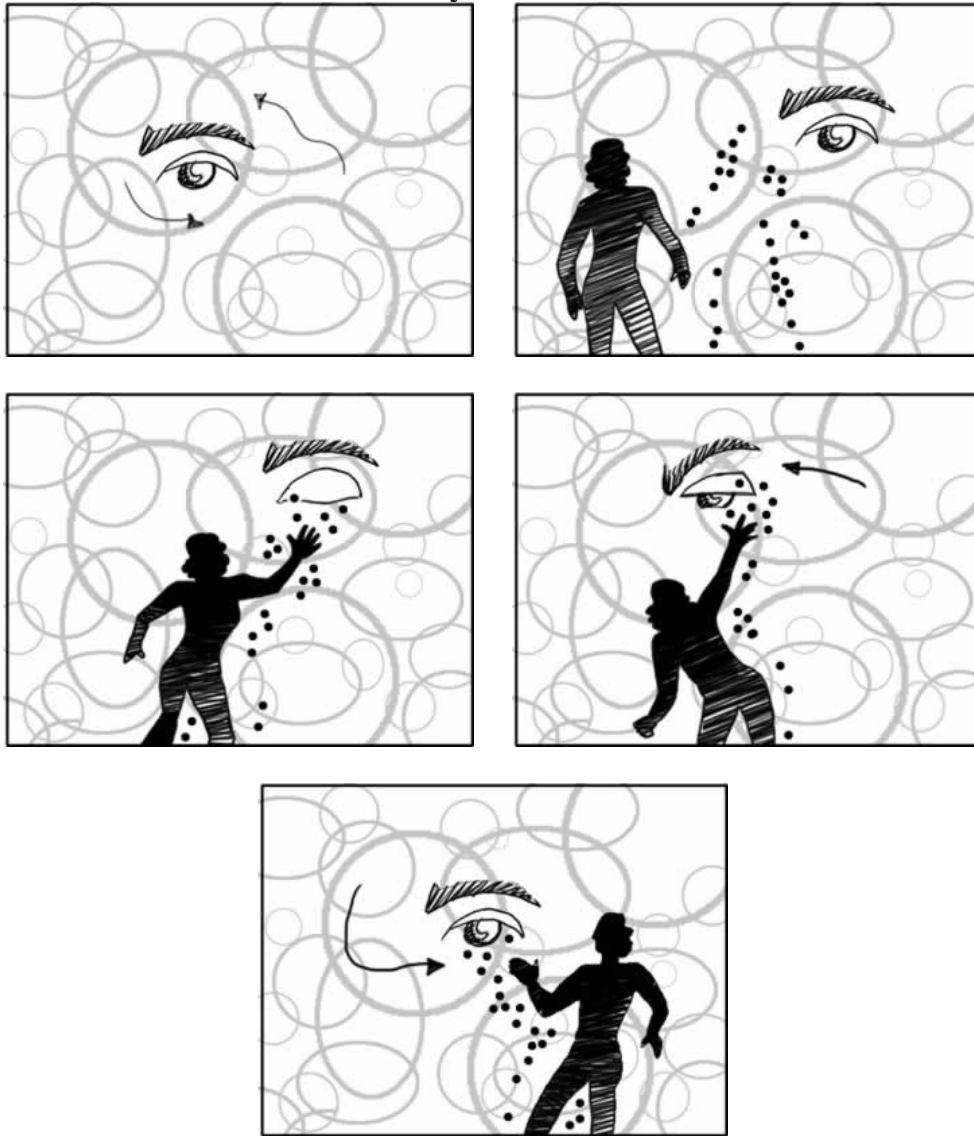
Linklater, Richard. "Waking Life" feature film. 2001. "interpolated rotoscoping"
programmed by Bob Sabiston.

Appendix C: Budget

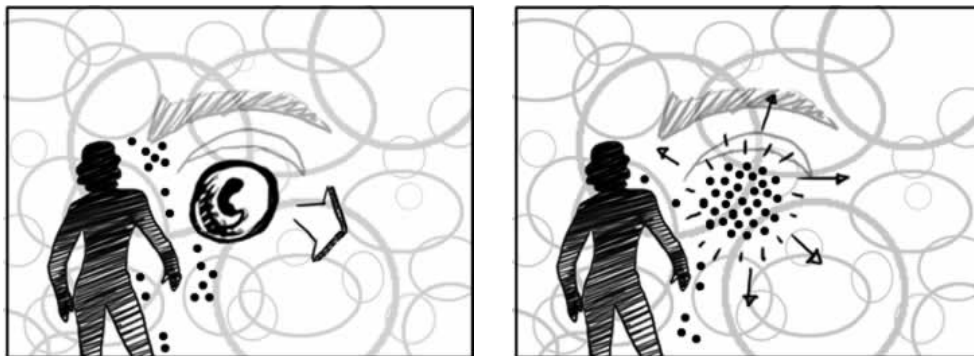
Item	Expense
Screen Frame Hardware	63.17
Screen Material	250.00
Reference Books	37.07
User Testing Supplies/Food	75.00
TOTAL	425.24

Appendix D: Final Storyboards

Eye Start



Transition



Dots



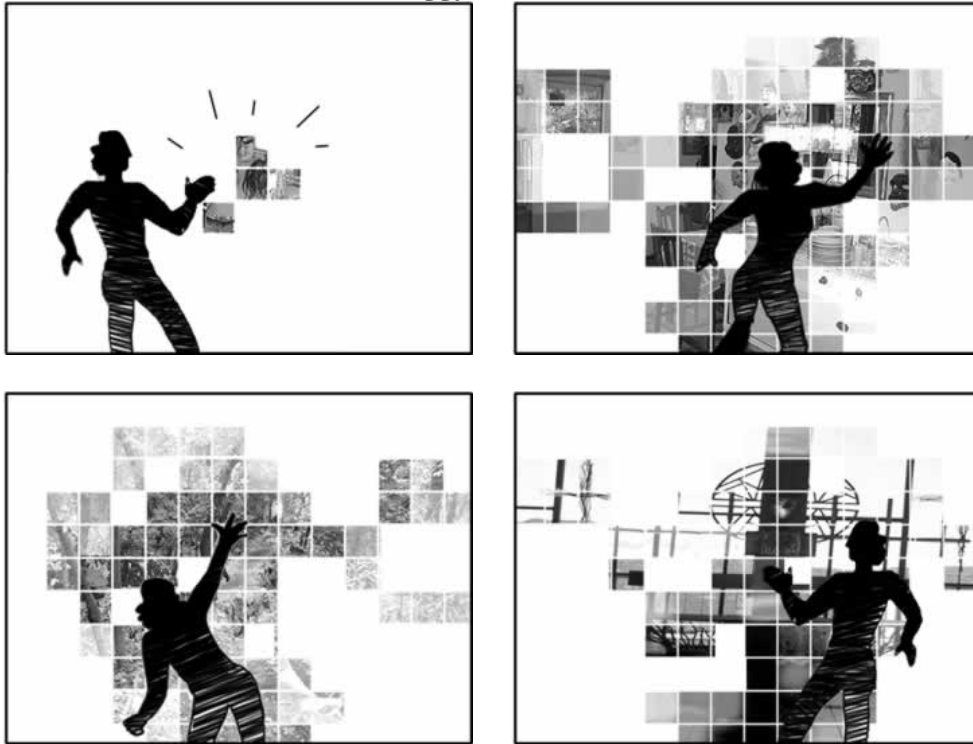
TinkerToys



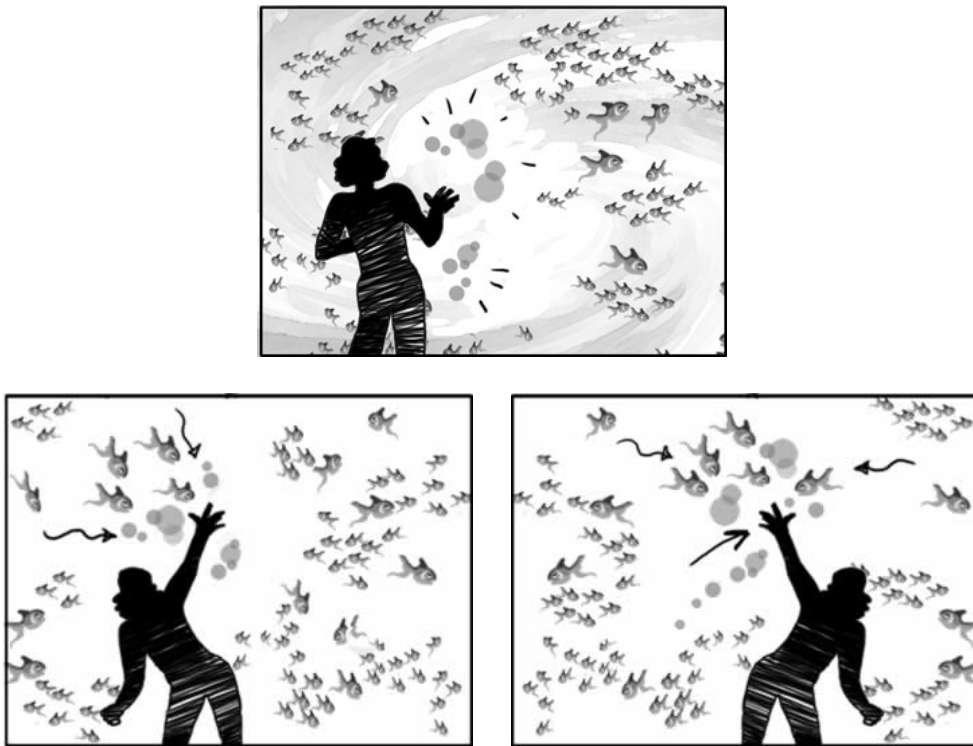
Transition



Foggy Windows



Fishes



Appendix E: Supporting Technology

EyesWeb

<http://www.eyesweb.org>

“The EyesWeb research project aims at exploring and developing models of interaction by extending music language toward gesture and visual languages, with a particular focus on the understanding of affect and expressive content in gesture.”

“EyesWeb has been designed with a special focus on the analysis and processing of expressive gesture in movement, midi, audio, and music signals.”

We chose the free EyesWeb platform to serve our visual processing needs. It features a visual programming interface with which users can assemble patches of software components to handle signal processing, much like the interfaces of more famous packages such as Max/MSP or Pure Data.

The Lucas-Kanade Feature Tracking Algorithm

http://www.ri.cmu.edu/pub_files/pub3/lucas_bruce_d_1981_1/lucas_bruce_d_1981_1.pdf

The heart of the capture side of Luminance is the EyesWeb implementation of the Lucas-Kanade Feature Tracking Algorithm. Details on this algorithm can be found in their initial publication:

An Iterative Image Registration Technique with an Application to Stereo Vision, Bruce D. Lucas and Takeo Kanade, the 7th International Joint Conference on Artificial Intelligence, 1981.

Funded by DARPA, the initial research was aimed at resolving spatial coordinates in 3D based on multiple camera viewpoints of the same scene. The algorithm scans for similar features in similar video frames and assigns tracking points to them, reassigning points as features are lost in movement. The EyesWeb implementation applies this algorithm to sequential frames in *time* rather than adjacent viewpoints in *space* – the net result is similar.

floc

<http://transmote.com/floc>

“floc is a communication gateway, written by [ben chun](#), that allows macromedia FlashMX to talk with any software that can understand UDP data.”

“floc is comprised of a java server and a suite of ActionScript functions. together, they allow communication between UDP-friendly software and FlashMX via flash's XMLSocket feature, which reads and writes TCP-formatted data. floc will run on any system that supports java and can run FlashMX, including win98/2000/XP, Linux, and OSX.”

floc is a highly critical component of Luminance’s technology, allowing data to flow from EyesWeb on our input computer to Flash on our output computer.

Macromedia Flash

<http://www.macromedia.com>

“Macromedia Flash MX 2004 allows designers and developers to integrate video, text, audio, and graphics into immersive, rich experiences that deliver superior results for interactive marketing and presentations, e-learning, and application user interfaces. Flash is the world’s most pervasive software platform, used by over one million professionals and reaching more than 97% of Internet-enabled desktops worldwide, as well as a wide range of devices.”

We chose Flash as our platform for creating and controlling the media of Luminance, due to our familiarity with the software and the subsequent ease of creating content. We were also previously licensed for its use, and didn’t have to pay for it.

Grant Skinner’s Proximity Manager

http://www.gskinner.com/blog/archives/2005/02/source_code_gri.html

We implemented Grant Skinner’s proximity manager to improve the efficiency of the range checking algorithms in our more complex scenes.

Various content creation software

Adobe Photoshop, Adobe Illustrator, Vegas Video, Final Cut Pro were all used to create the stills, animation elements and video of our digital content.

Appendix F: Misc. Technical Details

Open Sound Control

OpenSound Control ("OSC") is a protocol for communication among computers, sound synthesizers, and other multimedia devices that is optimized for modern networking technology and has been used in numerous application areas, including MIDI, audio, and video.

OSC Features:

- Open-ended, dynamic, URL-style symbolic naming scheme
- Numeric and symbolic arguments to messages
- Pattern matching language to specify multiple targets of a single message
- High resolution time tags
- "Bundles" of messages whose effects must occur simultaneously
- Query system to dynamically find out the capabilities of an OSC server and get documentation

Sockets

The key to connecting the Eyes Web Application to Flash through FLOSC is the idea of sockets. The following general description of a Socket was found on webmonkey.com:

"Every computer on the Internet talks to other computers on the Internet using sockets. When you open a connection to another computer, you use a socket. Email servers open a socket for every user checking his or her email and every user sending an email. Web servers open (and close) a connection for every file they transmit. (Note: Some web servers are smarter than this, but handwave it for the moment — we'll come back to it.) Webcams, Napster, and ICQ all run on sockets. Every conversation consists of at least two: one on each end. If you're going to write applications which communicate over sockets, you'll have to deal with each of those ends. Even if you're only planning to deal with one side, it's still a socket."

Key Flash Classes and Methods

This section introduces some of the main Flash classes, methods, and event handler used in connecting Flash to the FLOSC gateway

XMLSocket Class

ActionScript provides a built-in XMLSocket class, which lets you open a continuous connection with a server. A socket connection lets the server publish, or *push*, information to the client as soon as that information is available.

Without a continuous connection, the server must wait for an HTTP request. This open connection removes latency issues and is commonly used for real-time applications such as chats. The data is sent over the socket connection as one string and should be formatted as XML. You can use the XML class to structure the data. To create a socket connection, you must create a server-side application to wait for the socket connection request and send a response to the SWF file. This type of server-side application can be written in a programming language such as Java.

The XMLSocket class implements client sockets that let the computer running Flash Player communicate with a server computer identified by an IP address or domain name. The XMLSocket class is useful for client-server applications that require low latency, such as real-time chat systems. A traditional HTTP-based chat solution frequently polls the server and downloads new messages using an HTTP request. In contrast, an XMLSocket chat solution maintains an open connection to the server, which lets the server immediately send incoming messages without a request from the client.

To use the XMLSocket class, the server computer must run a daemon that understands the protocol used by the XMLSocket class. The protocol is described in the following list:

- XML messages are sent over a full-duplex TCP/IP stream socket connection.
- Each XML message is a complete XML document, terminated by a zero (0) byte.
- An unlimited number of XML messages can be sent and received over a single XMLSocket connection.

XMLSocket.onConnect Method

Example usage: myXMLSocket.connect(host:String, port:Number) : Boolean

Parameters

Host: String; a fully qualified DNS domain name or an IP address in the form *aaa.bbb.ccc.ddd*. You can also specify null to connect to the host server on which the SWF file resides. If the SWF file issuing this call is running in a web browser, *host* must be in the same domain as the SWF file; for details, see [Description](#).

Port: A number; the TCP port number on the host used to establish a connection. The port number must be 1024 or greater.

Establishes a connection to the specified Internet host using the specified TCP port (must be 1024 or higher), and returns true or false, depending on whether a connection is successfully established. If you don't know the port number of your Internet host computer, contact your network administrator.

XMLSocket.OnXML Event Handler

```
myXMLSocket.onXML = function(object:XML) {  
    // your statements here  
}
```

object An XML object that contains a parsed XML document received from a server.

Invoked by Flash Player when the specified XML object containing an XML document arrives over an open XMLSocket connection. An XMLSocket connection can be used to transfer an unlimited number of XML documents between the client and the server. Each document is terminated with a zero (0) byte. When Flash Player receives the zero byte, it parses all the XML received since the previous zero byte or since the connection was

established if this is the first message received. Each batch of parsed XML is treated as a single XML document and passed to the onXML method.

FLOSC XML Data Type Definition (DTD)

```
<!ELEMENT OSCPACKET (MESSAGE+)>
<!ATTLIST OSCPACKET
    ADDRESS CDATA #REQUIRED
    PORT CDATA #REQUIRED
    TIME CDATA #REQUIRED
>

<!ELEMENT MESSAGE (ARRAY | ARGUMENT)* >
<!ATTLIST MESSAGE
    NAME CDATA #IMPLIED
>

<!ELEMENT ARRAY (ARRAY | ARGUMENT)*>
<!ELEMENT ARGUMENT EMPTY>
<!ATTLIST ARGUMENT
    TYPE (i|f|h|d|s|T|F|N|I) "i"
    VALUE CDATA #IMPLIED
>
```

Sample FLOSC Packet

```
<OSCPACKET ADDRESS="127.0.0.1" PORT="1080" TIME="0">
  <MESSAGE NAME="point_0">
    <ARGUMENT TYPE="f" VALUE="321.44025" />
    <ARGUMENT TYPE="f" VALUE="222.42276" />
  </MESSAGE>
</OSCPACKET>
```

Luminance ActionScript Pseudocode

Version 1

Define an XML Socket Connection

Create Handler for Connecting

Create Handler for failed Connection

Create Handler for closing Connection

Create Handler for using incoming XML data through socket connection

Current Version

- XML Socket Definition
- Handler function for Connection Success, Connection Failure, Connection Closure
- If Incoming XML, then Parse XML into X,Y,label “siblings”
 - Create variables for x coordinate and y coordinate
 - Create empty movie clip on new level defined by incoming label data
 - Move movie clip to X,Y values
 - Determine random value for object radius and transparency
 - Pass radius, and transparency values to draw circle function
 - Socket listener is continuously waiting for incoming XML data

Appendix G: User Testing forms



PROJECT LUMINANCE EVALUATION

A Cal State East Bay Multimedia Graduate Program Thesis Project 3/4/05

Please choose one opinion for each statement.

This was easy to do. Yes ___ Somewhat ___ No ___ Couldn't tell ___

This was fun to do. Yes ___ Somewhat ___ No ___ Couldn't tell ___

I got lost in what I needed to do. Yes ___ Somewhat ___ No ___ Couldn't tell ___

Things happened too fast. Yes ___ Somewhat ___ No ___ Couldn't tell ___

I'd like to do another one of these. Yes ___ Somewhat ___ No ___ Couldn't tell ___

It went on too long. Yes ___ Somewhat ___ No ___ Couldn't tell ___

+++++

What was your favorite part?

Horizontal lines for writing the favorite part.

If there were three things you could change, what would they be?

1.) _____

Horizontal lines for writing the first change.

2.) _____

Horizontal lines for writing the second change.

3.) _____

Horizontal lines for writing the third change.

PROJECT LUMINANCE EVALUATION

Additional comments, please:

Thanks for your participation. Next time we'll have chocolate.

Appendix H: User Testing Consent Form



PROJECT LUMINANCE TEST

A Cal State East Bay Multimedia Graduate Program Thesis Project

Purpose

In order to effectively evaluate the responses to our art installation, we need to videotape the sessions. We would like to obtain your consent to videotape your usability testing session today. The tape will be used only internally for our thesis project. It will not be broadcast, distributed, or used for any other purpose. America's Funniest Home Videos didn't offer enough.

Consent

I, the undersigned, hereby give my permission for today's usability session to be videotaped for the purposes described above.

Name: _____

Signature: _____

Date: _____

MONITORING TESTS LIVE

Videeing and monitoring notes will give us best bang for buck re time & schedule.

Most instructive or all videos can go up on site.

Each user plays without having observed others.

No revealing of project's content/usage by finished users to upcoming ones.

Instructions = only ask them to face the screen and tell us aloud what they're thinking/feeling as they go through the process.

No direct answering of questions during play but can make encouraging response if asked – don't want clinical atmosphere.

No voluntary comments to users while they are playing.

USER EVALUATION ELEMENTS:

Pleasure
Puzzlement
Frustration
Feedback response/ understanding
Assertiveness
Immersion/interest

INSTALLATION EVALUATION FACTORS

Errors
Response time
Visual clarity

