

# Attention: An Experimental Film Installation

## <http://www.theexperimentalfilm.com>

M.Sc. s0789671: Parag K. Mital

April 17, 2008

## 1 Introduction

This installation investigates how eye-tracking can be represented in the form of an interactive film editor to create a new output film. The eye data recorded from a first person perspective film<sup>1</sup> directs the video and audio in an immersive space while the audience is overwhelmed by the surprising nature of the output film<sup>2</sup>. Multiple angles of a scene are recorded to provide wide-angle (WS), mid-to-wide angle (MS), and close-up (CU) shots. The fourth angle is taken as the first-person (FP) perspective. A composite video of 4 narratives in FP are stitched in a 2x2 array to create the eye-tracked film. This video was shown to 20 people while they were being eye-tracked for screen position, pupil dilation, and blink data. The additional angles of the scene are not shown to the eye-tracked users as these are used to “reward” the eye-tracked user with a deeper revelation of the narrative that pulls their attention.

The display of the original eye-tracked film as a composite of 4 narratives is inspired by Michael Figgis’ “Timecode” which has 4 concurrent narratives in a loosely scripted plot centered around an earthquake and dramatic love story. The nature of our film is inspired by Jorge Luis Borges who often used the idea of circularity in time. The idea that a circular “labyrinth” of film edits that correspond to a narrative which is inherently circular in nature would lead to some interesting developments. The narratives themselves are separated into two corresponding lovers, Man and Woman, and Sara and Josh. These two lovers share much in common and find themselves struggling through desire. They eventually meet in person, in another metaphysical world, or as themselves, depending on how one interprets the story. The idea of desire, anticipation, and the marriage of the two narratives are inspired by William Blake’s, “The Marriage of Heaven and Hell”:

Those who restrain desire do so because theirs is weak enough to be restrained; ... And being restrain’d, it by degrees becomes passive, till it is only the shadow of desire.

---

<sup>1</sup>denoted as “the eye-tracked film” from here on.

<sup>2</sup>denoted as “the output film” from here on.

## 2 Previous Work

Some similar work in the realm eye-tracking as a method of interactivity has emerged from the Tisch School of the Arts in NYU. In the show entitled, “Peepholes”, users can look through keyholes into living rooms. The voyeur’s eyes are then eye-tracked and this output is used to control a camera inside the room. The output of this camera is shown on a television for people inside the room to see, propogating the feeling of paranoia of being watched.

White Glove Tracking is a project which used manually segmented images of a video to locate Michael Jackson’s white glove during a concert performance. This database has led to a number of projects such as the Paul Pfeifer Tribute, which uses the database in order to center a video based on the location of the white glove.

An interesting work by Golan Levin and Greg Baltus uses the idea of the artwork looking at the audience. In the installation entitled, “Opto-Isolator”, a single mechanical human-sized eye responds to the gaze of its audience with most unnerving behaviors. For more information on Golan Levin and this particular installation, visit his website at <http://www.flong.net/>.

## 3 Platform

The task of representing eye-tracking data as an interactive film editor called for multiple timelines that could be synchronized in real-time and edited based on a stream of incoming data. Thus, after looking into the advantages, the platform decided upon was Cycling ’74’s, MaxMSP + Jitter Library (<http://www.cycling4.com>). This coding platform allows for real-time manipulation of parameters, OSC for communicating, and powerful DSP routines for audio processing. Additionally, with the Jitter Library, many OpenGL and GPU operations are conveniently exposed that are necessary to edit the film in real-time.

## 4 OpenSoundControl

Because of the enormous amount of data that needs to be processed in real-time, it is necessary to distribute processing amongst 3 computers. Although it is able to run on a single computer, the frame rate averages about 12 fps. With two additional computers, the processing is equally distributed as: one computer for audio and eye-tracking processing, a second computer for video processing of 2 of the four narratives, and a third computer for video processing of the remaining 2 narratives. To coordinate timing messages between each computer, OpenSoundControl (OSC) developed by Matt Wright at the University of California in Berkeley makes use of the stream capable nature of UDP to encode and decode packets creating a friendly framework for routing messages between computers. This library is freely available from the Center for New Media And Technology (CNMAT) website located at <http://www.cnmat.Berkeley.edu>.

## 5 Eye-Tracking Data

Though the Eyelink 1000 (SR Research) is capable of 1000 Hz recording, the framerate of the film and the naïve manner in which we use fixation limits the need to only 25 Hz. For each recording, the eye-tracker reveals the (x, y) location onto the screen, the pupil dilation, and whether there was a blink (i.e. no pupil present). The experiment is setup on a 20" CRT Monitor at 120 Hz refreshrate and 120 DPI. The 960x540 movie is centered onto the 1024x768 resolution screen. A Perl script written by Dr. Tim J. Smith extracts the essential data from the ascii version of the Eyelink Data to a file that is easily parsed as: [frame — x — y — pupil dilation]. Using this data, 4 measures are taken per frame: source dimension, fixation, blink, and pupil dilation. This can be seen in the second column of the Figure-3.

### 5.1 Source Dimension

Since the film are arranged in a composite of a 2x2 array, knowing which film an eye-tracked user is currently watching is a simple calculation based on the center coordinate of the screen. Each film is assigned a unique number from 1 - 4 with 1 being the top-left film, and 4 being the bottom-right film. This is shown in Figure-1. The variables (cx, cy) depict the center coordinates of the screen. Thus, for a 1024x768 screen: (vx=512, vy=384).

### 5.2 Fixation

Rather than implementing fixation as defined in visual cognition text, fixation for the interactive film editor is defined

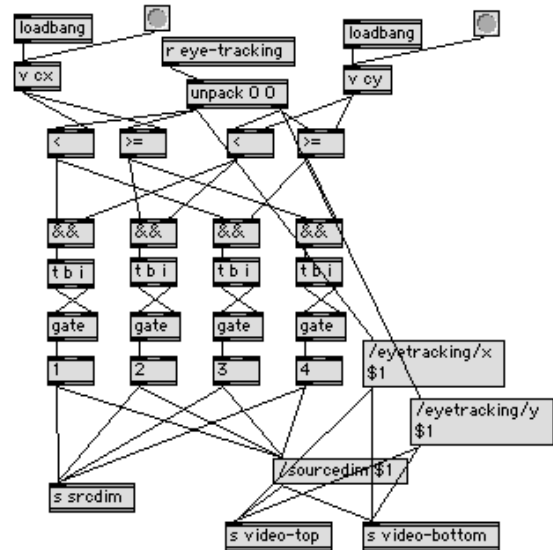


Figure 1: This subpatch computes a number from 1-4 signifying which of the four narratives the user is watching.

by the entire region of one of the four films. An actual measure of fixation would entail measuring the change in degrees or a visual angular velocity corresponding to a change of greater than 1 degree. Though, the nature of the output of the film is already so erratic that this measure would have only further overwhelmed the audience. A more suitable measure would have taken pursuit movements rather than fixations, where the durations correspond to how long the eyes pursue an object during the dynamic image. The remainder of this document will use the term fixation as the naïve measure of staying within one of the four source narratives, though the reader should note that fixation is much more intricate than our implementation. Thus, similarly to the source-dimension calculation, knowing the screen resolution and the film dimensions, it is trivial to calculate fixation. Between each frame at PAL rate, fixation accumulates 40 ms as long as the region of the film stays the same. This is shown in Figure-2.

### 5.3 Blink

Because the data from the eye-tracker is pre-processed by Tim J. Smith's Perl script, blinks are represented by the (x, y) coordinates as (-, -). Thus, knowing when a user blinks is a binary measure. This is used to trigger a Max gate in the video and audio, essentially triggering an effect.

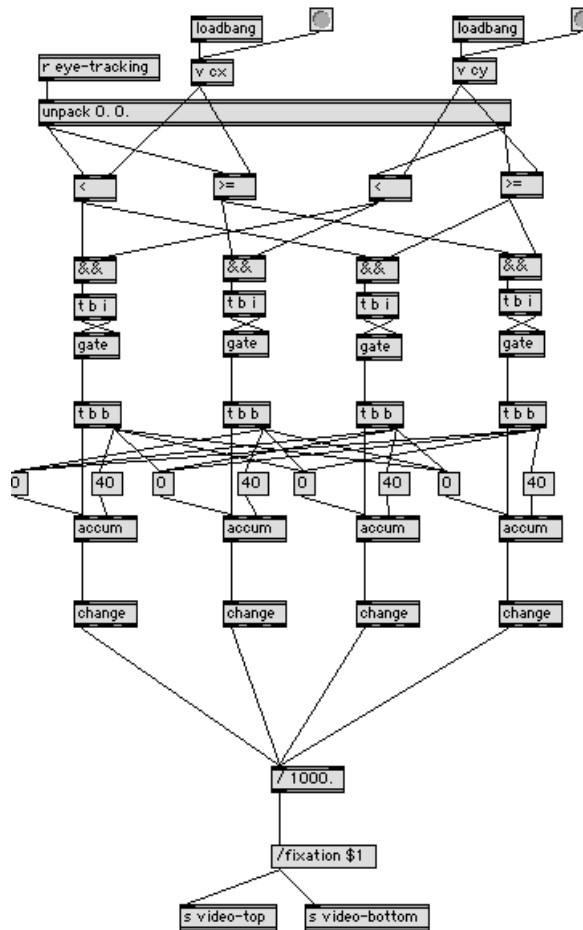


Figure 2: This subpatch calculates fixation onto a region defined as the entire film.

## 5.4 Pupil Dilation

The final column of the eye-tracking data details the pupil dilation. There is no actual “unit” representation for this value. Rather, it is a measure that is computed by the Eye-link. Also, its range spans a different amount for each eye-tracked user that is often in the range of 1000 - 8000 units. Since there is such a difference amongst each user, the data is pre-processed to include on the first line a minimum pupil dilation and a maximum pupil dilation. The frame’s pupil dilation is then sent directly to the audio and video portions of the program to control a parameter of feedback and saturation respectively.

## 6 Audio

With 4 narratives, at length of 9 minutes each with 3 different angles (WS, MS, and CU), along with 4 fixed audio tracks for each of the 4 narratives, there is at its base, 16 tracks of 9 minutes in length that need to synchronize and play back at PAL rate (25 fps). Additional to this is a database of nearly 60 voiceovers (short clips of audio nearly 2 seconds in length), for each of the 4 narratives that should be triggered by eye-tracking data, giving a total of 240 additional buffers of audio that can be triggered at any time. How could eye-tracking data accurately coordinate all of this data while still realising that goal within the limits of real-time CPU and GPU processing?

### 6.1 Fixed Audio

After much experimentation and help from the Cycling ’74 community (thanks deKam), the solution to synchronizing all of this data without killing the CPU is to have 1 master audio track as the timecode for everything else. To begin, the time scale is gathered from one of the films (assumed to be the same across all films). This obviously could have been hard-coded at 600 for our particular library of films. Though as a modular program, it makes more sense to leave this as a simple dependency. The file position of the audio track is multiplied by the timescale to get the time in milliseconds. Dividing by 1000, this time is sent to each of the films to output the current video frame for that given time. This process is shown in Figure-5.

The original eye-tracked film has 8 channels or 4 mono channels of audio corresponding to the 4 narratives. This is the fixed audio tracks for the both the eye-tracked and the output film.

### 6.2 Variable Audio

In the original eye-tracked film, there is only fixed audio. Adding to the output film, a large database of nearly 60 voice-overs for each of the narratives depicting the internal thoughts of the character at the given point in the story is triggered by the eye-tracking data. Thus, an initial database details when the voiceovers are allowed to be triggered, detailing the name of the audio track, the starting time when the track is allowed to be played, and the duration of when this track is allowed to be played, effectively detailing the ending point. These details are parsed in the patch shown in Figure-6 to create a frame based real-time parsable script that will simply detail the audio file for each of the 4 characters.

Using the output from this patch, the line number is sent to the text file to gather at any given time the corresponding file name of the voice over file to be played as shown in

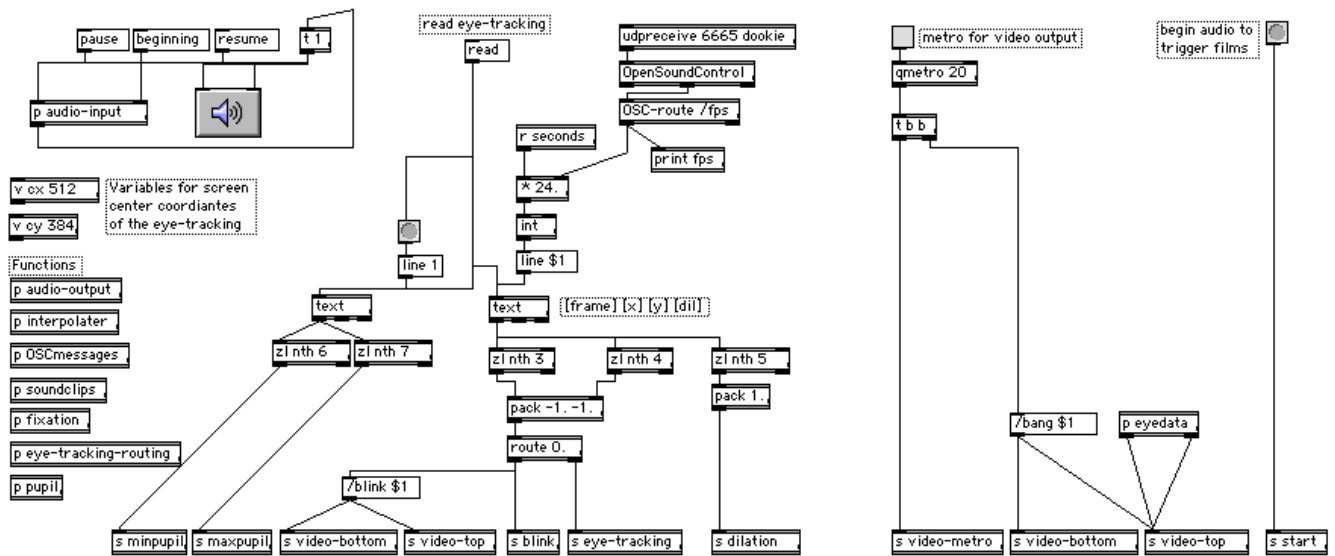


Figure 3: The topmost level of the audio and eye-tracking processing patch.

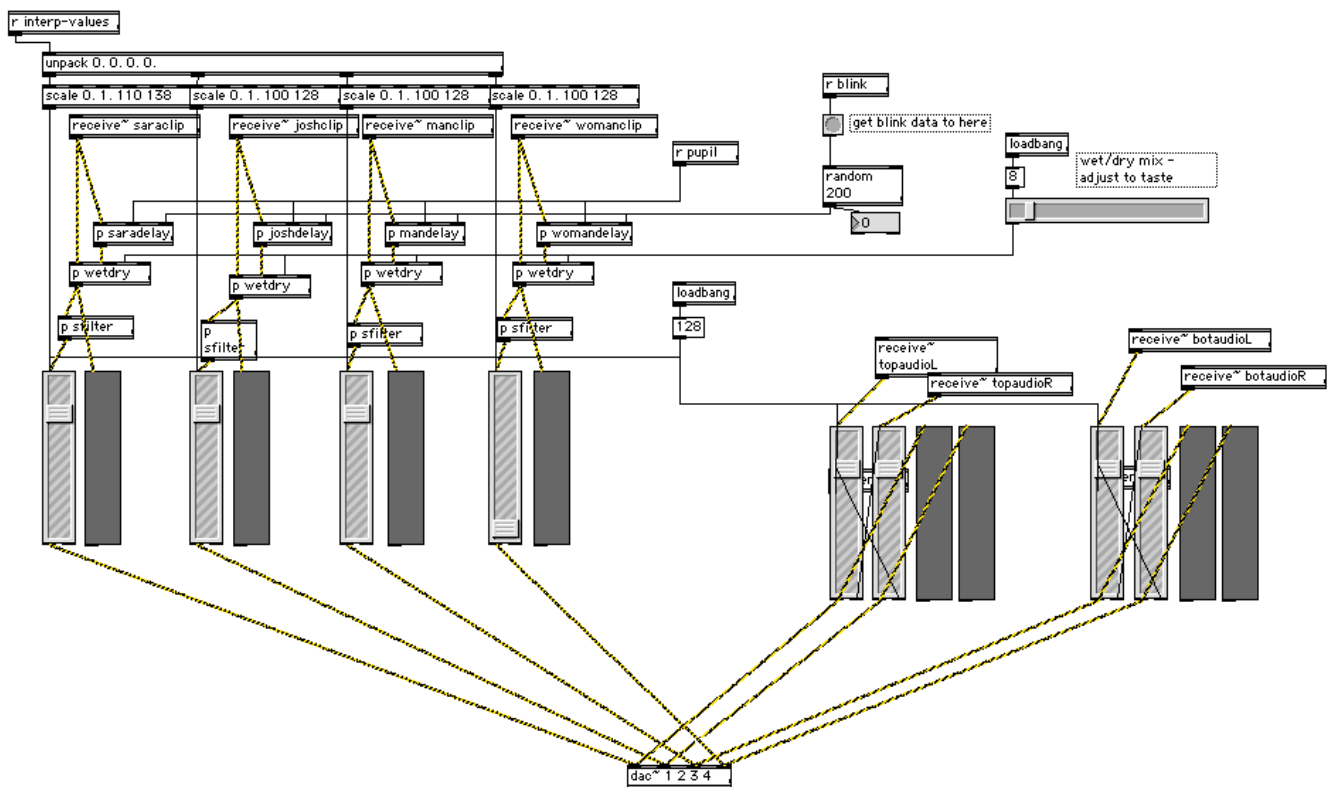


Figure 4: This shows the actual output portion of the audio. Fixed audio is on the right side while the soundclips are on the left. The signal faders' volumes are also controlled by the interpolating weights from the interpolate subpatch shown in Figure-9.

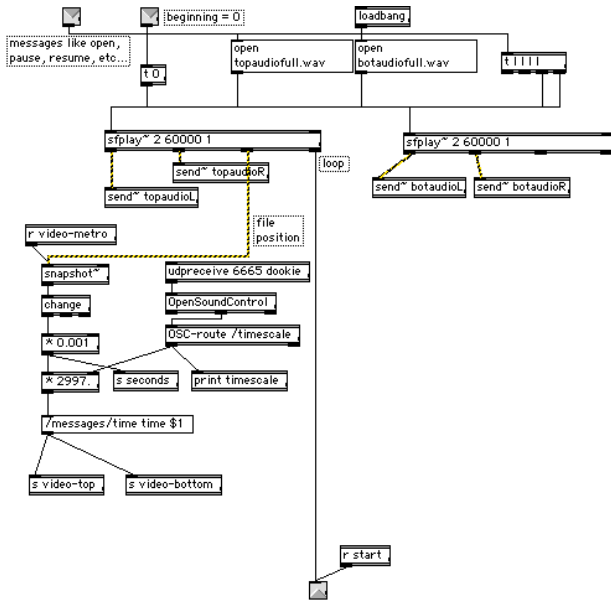


Figure 5: The fixed audio tracks for the 4 narratives are combined as 2 stereo tracks. Additionally, the time messages are coordinated by one of these tracks and the messages are sent to each of the films.

Figure-7. Then, by knowing the source dimension, a simple Max-gate controls which of the four text files are triggered. With the file name, this is sent to a buffer to be output (see Figure-8). Additionally, note how the gate just before this buffer ensures that only 1 sample can be output at any given time. This ensure that the current sample has to finish before a new sample can be played.

### 6.3 Dynamic Audio

Keeping the audio fairly minimal for the eye-tracked film, the eyes are not so directed by the sound but rather the content. However, for the output film, as the processing takes into account the eye-tracking, the variable audio added to the fixed audio is an amalgamation of cues for direction. To achieve this direction, the discrete spatialization inherent in the 4 channels of output begin to orient the audience. To further interpret the spatialization a more continuous domain (e.g. imagine 4 speakers simulating what 32 speakers arranged in a circle could achieve), interpolating signal faders based on the spatial location of the eye-data simulates a 2-D field of dynamic spatial audio for the physical space of the installation. Using a javascript program written by Ali Moment (see aLib from <http://alimomeni.net>), using the eye-tracking data to assign signal levels for the audio is fairly straight forward. First, weights are assigned for 1-4 in the space of (-1, -1) to (1, 1). The eye-tracking data is

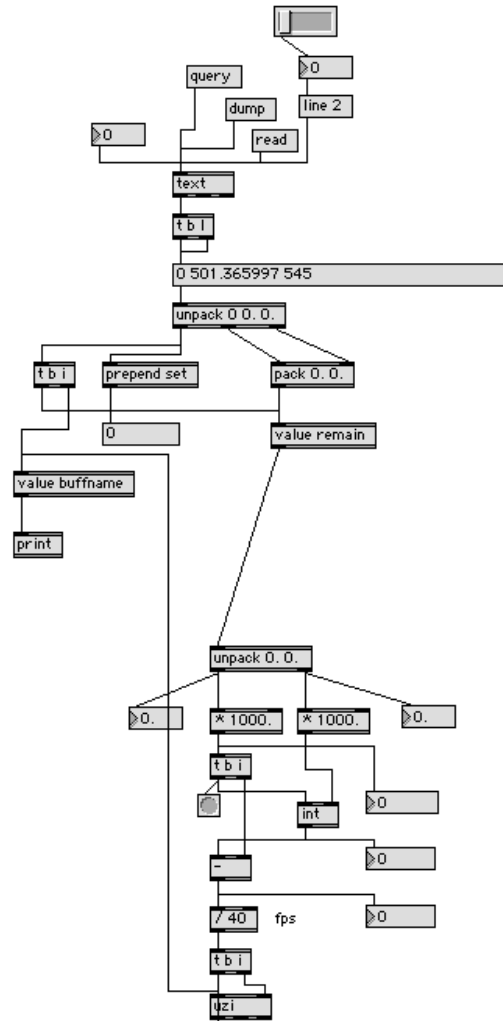


Figure 6: A database of about 60 sound clips and the corresponding start times and durations they can be triggered are parsed to create a database for each frame of the video corresponding to a single audio track.

then scaled to this space. The number are sent to the interpolating space, and the corresponding Gaussian weights are output in a list as shown in Figure-9. These numbers are then sent to the signal faders as shown in Figure-4.

## 7 Video

Given three angles of a scene for 4 narratives, how does eye-tracking interactively edit the film?

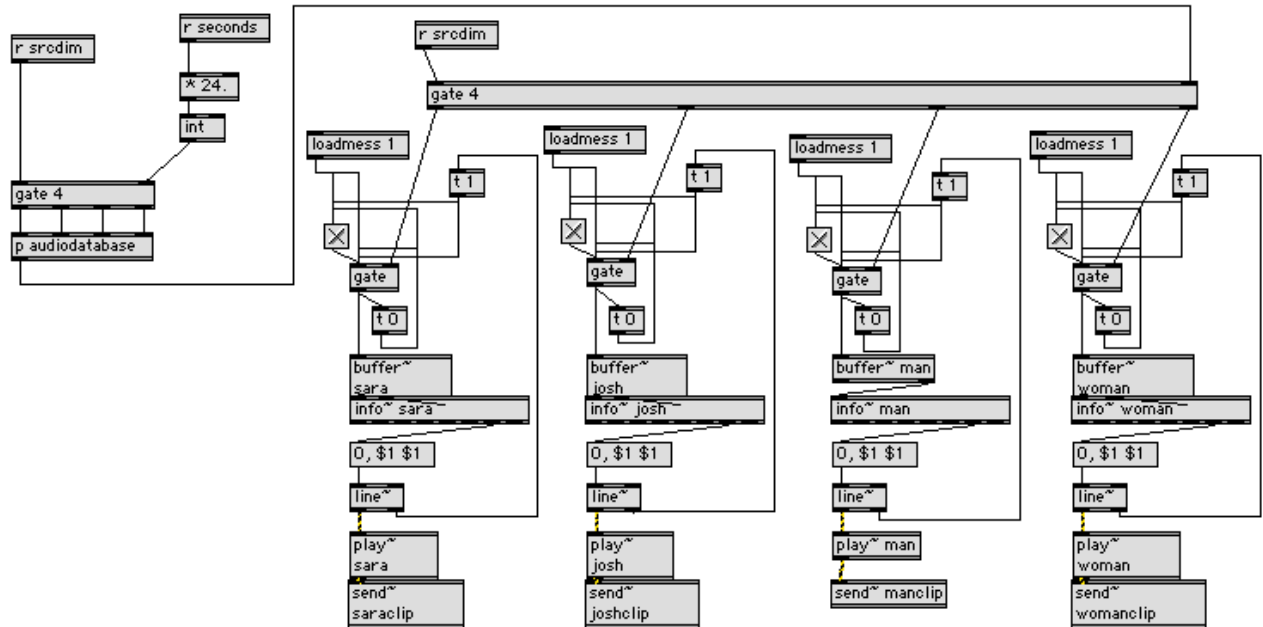


Figure 8: The source dimension controls the output of the gate which triggers a file name in the samples database. This file name is sent to be played in the corresponding buffer.

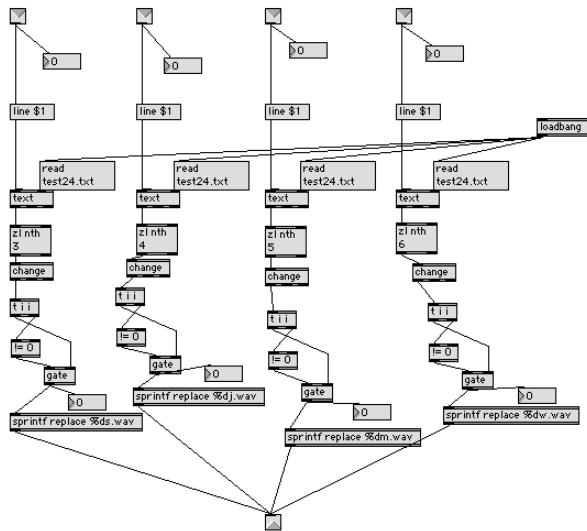


Figure 7: The database created from Figure-6 is parsed in realtime by this patch to gather the triggered sound clip's file name.

## 7.1 Heaven and Hell

To begin, the space of projection must be considered. The installation space is an immersive space with 2 back-projected surfaces encasing an audience in a space of nearly 20 square meters. The adjacent sides of the projected surfaces are mostly closed with a small 1 meter gap for an entrance to the space. Thus, the atmosphere is meant to encapsulate an entire audience to further the idea of the eyes as the director and that the audience is envisioning what the eyes have created.

As there are 2 projector surfaces, the first idea is to combine the top two narratives, Sara and Josh (SJ), onto one projector, with the remaining two, Man and Woman (MW), on the other projected surface. To represent the ideals inspired by William Blake and the Marriage of Heaven and Hell, each surface is processed by a pixel shader. This is easily accomplished by moving the processing onto a jitter slab (jit.gl.slab). Once, the stream of computation is created, a programmable shader performs operations onto each pixel with the speed of the GPU.

For the innocent young couple SJ, a bleach-bypassed pixel shader written by vade (<http://abstrakt.info>) presents a feeling of lost desire, young love, antiquity, and low saturation. For MW, a different tone entirely of tinted technicolor represents a beating flame, the hell of eternal desire, and in-

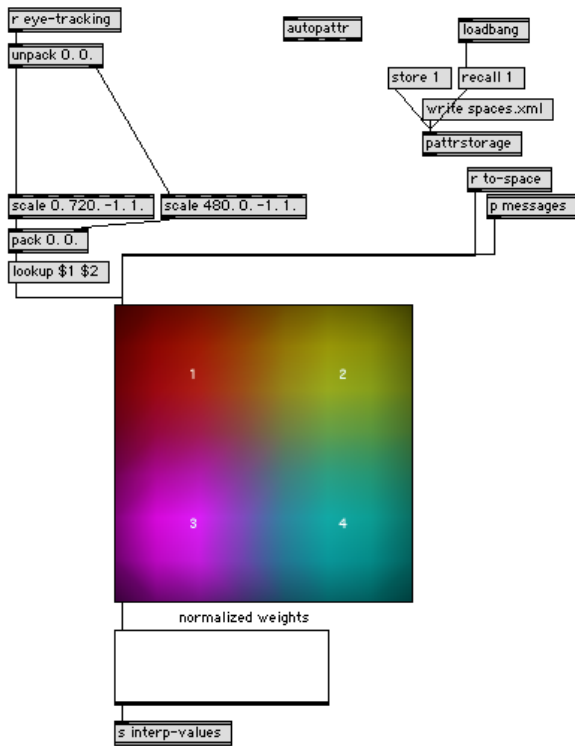


Figure 9: Using the values of the (x, y) eye-tracking coordinates, Gaussian weights are output from Ali Momeni’s JSUI object from aLib..

herently, marriage itself. Ultimately, it is up to the audience to make any interpretations.

## 7.2 Fixation

As there are 4 narratives each with 3 angles, WS, MS, and CU, there is a lot of editing to be done interactively. Fixation onto one of the four narratives determines if there is any output. That is, if there is eye-data for either Sara or Josh, then SJ’s projected surface will show the corresponding film represented by their fixation (see Section-5.2 for how this is calculated). As long as the user had been watching Sara or Josh, the audience will see SJ’s projected surface displaying a story. Once the eye-data moves onto either Man or Woman, SJ’s projected surface pauses, while MW’s story jumps to the current point in time. Thus, the film cuts between narratives are determined by simply watching that narrative. Once the user stops watching that narrative, it pauses only to fast-forward to when they come back to watch it again. This is accomplished by the master timecode kept by the audio track, and a gate controlled by the value of Source-Dimension discussed in Section-5.1 to con-

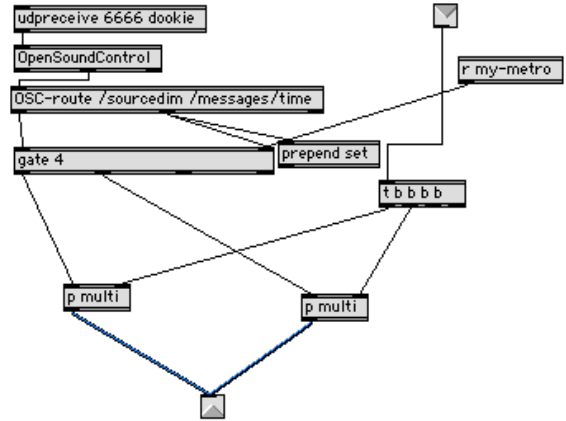


Figure 10: This sub-patch controls the metronome to trigger only the film that the eyes are currently watching.

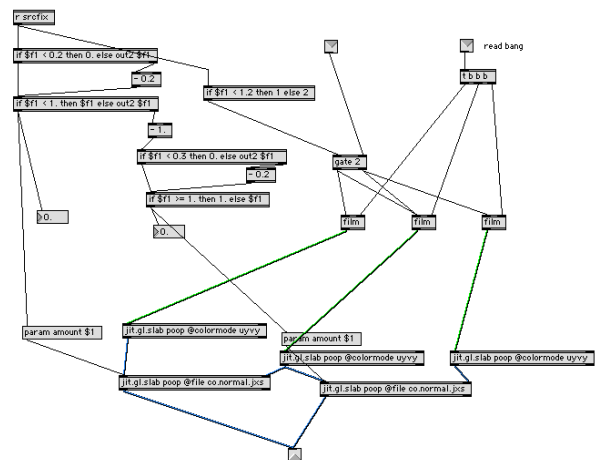


Figure 11: This sub-patch uses fixation data to apply the crossfade between WS, MS, and CU films.

control the metronome output. See Figure-10 for this process.

To begin with, all narratives are displayed as a WS. That is, when eye-data arrives for a single film (i.e. Source Dimension), its corresponding WS is shown. As their attention (i.e. fixation value) accumulates for the same narrative for longer than 300 ms, the film will start to crossfade into the MS, enticing the audience to see more. As 1000 ms comes, the MS stays in clear focus up until 1300 ms. And again, as attention stays fixated even longer, from 1300 ms until 2000 ms, the MS dissolves into the CU, to reveal a clear focused CU of what the eyes were attracted towards. See Figure-11 for the max-patch of this process.

### 7.3 Blink

To represent a blink, a Gaussian filter performed 6 times effectively blurs the entire image for the duration of the blink. As the rate of blinking can depend on age, fatigue, and medicinal activity, blink data can often vary among the different collected eye-data, with some blinks as long as 15 frames (600 ms), and as short as 1 frame (40 ms). Though on average, blinks tend to take as long as 300-400 ms. Further, the actual quantity of blinks varied across the data to as minimal as 7 sporadic blinks for the entire 9 minutes to as much as a 10 blinks in a single minute. Thus, the actual output of the blink may not be very noticeable unless one is paying attention very closely to the details. Further, the nature of the gauss sheets already provides an aliased distortion that could be interpreted as a Gaussian filter itself. Thus, the nature of the blink is a subtle fact that is only meant to keep the video from “drying out.”

### 7.4 Pupil Dilation

To account for the pupil dilation, one last shader controls a simple measure of brightness, contrast, and saturation. These values are mapped to a scaled version of the pupil dilation to create a slight variation in overall exposure. Thus, among different eyedata, the overall exposure differences can account for a fairly dark interpretation of the film or rather highly saturated interpretation.

### 7.5 Director Name

As part of the installation, each director’s name (eye-tracking data) appears at the bottom of the output film. To achieve this, a database hard-coded in MaxMSP sends text via OSC to the video computers. This is also controlled by the Apple Remote (aka.appleremote by Masayuki Akamatsu <http://www.iamas.ac.jp/~aka/max/>) to allow skipping between directors in the middle of the film or by starting from the beginning again. As the output of the film finishes, the next director’s output film is automatically triggered. This patch is shown in Figure-12).

### 7.6 Eye-Tracking Display

Though not part of the installation, effort was made to try to incorporate a visual smearing effect based on the actual fixation sites and saccades of the eyetracking. The matrix for computing this is shown in Figure-13. This matrix shows the actual eye-tracking in real-time and can be sent as the alpha channel with a jit.gl.videoplane set to “@blendingenabled 1” to see an actual smearing effect with the eyes controlling the effectively displayed pixels. However, the essence of the installation was focused towards an actual

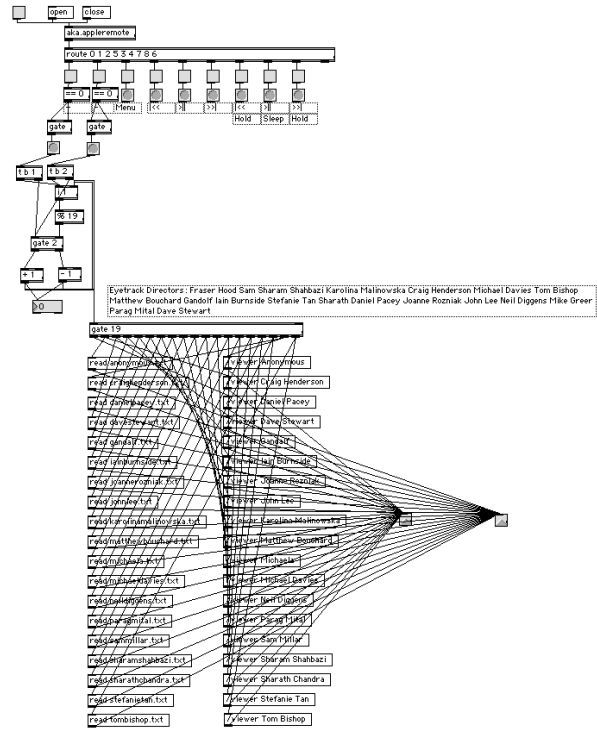


Figure 12: Masayuki Akamatsu’s aka.appleremote controls the playback. Also, a small database of the director’s names are sent via OSC to output a textual display of the eye-tracking director’s name.

film rather than an eye-tracking display. And thus, the representation of simply showing the fixation sites and saccades was not representative enough of the actual experience to have it a part of the film editorial process.

## 8 Discussion

In Michael Figgis’ Film, “Timecode,” audio levels are often adjusted to *direct* the viewer to a particular narrative of importance. The original eye-tracked film does not have any kind of audio manipulation to direct the eyes. Rather, it wants the eyes to direct a film based on whatever pulls their attention. Thus, in the output film, the eyes assign weights to the signal faders, controlling the audio levels like Michael Figgis’ “Timecode” had done. Rather, because of the nature of gaze, the eyes are constantly in a rapid search. Add that fact to the nature of watching four narratives simultaneously, and the audio will have some interesting and of course overwhelming direction.

It is interesting to see how an audience watches the output film. As its direction is led by the eye-tracking data, what if the audience member wants to listen to something

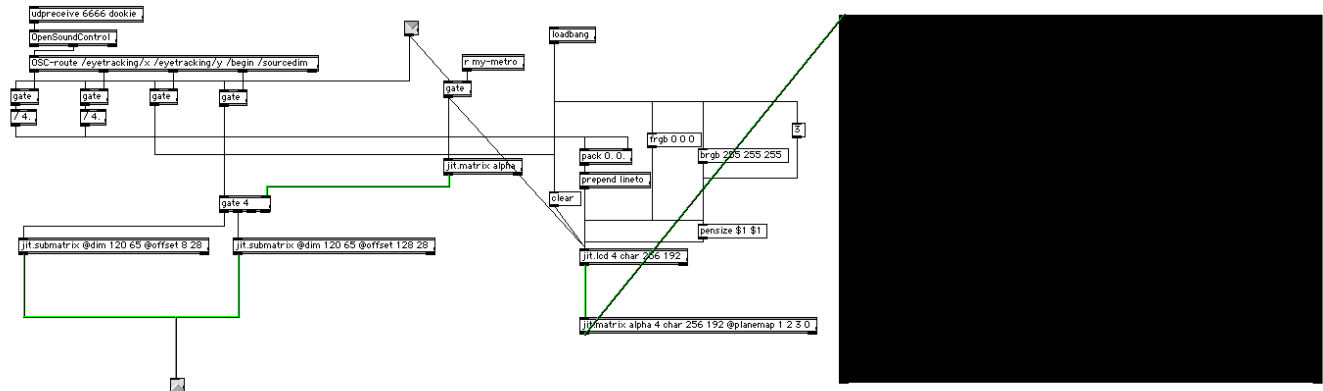


Figure 13: This sub-patch receives the eye-tracking data in real time from the audio/eye-tracking computer via OSC and creates a visual display in the matrix, “Alpha”, which could be used either to display the eye-tracking directly, or to alpha-blend/smear the output film.

that the eye-tracked user didn’t find particularly interesting? That something grabbed an audience member’s attention and yet did not the original director. And similarly for the video output. The audience member may be captivated by a narrative of Sara and Josh. However, the director was pulled to the Man and Woman and so the narrative of Sara and Josh stops. The audience may either feel regret that they will not know what happened, or rather, feel that something really interesting must be happening to the other narrative. What captured their attention?

Also, some interesting effects result from ensuring the voice-over clips finish. For example, the clip can be triggered, while the eye-data immediately saccades to the opposite side of the screen. Because of the interpolating audio weights, this has the effect of a strong attack and very quick delay. However, imagine now that the eye-tracking data comes back to the narrative which triggered the sound clip. Then, the gain for that channel will increase once again. These subtle effects give an audience a lot to think about when it comes to how the eye-tracking data has been represented in the real-time output film<sup>3</sup>.

<sup>3</sup>For a discussion on the other audio effects, see <http://www.theexperimentalfilm.com/>