ORGANIC SOUND
A Living Soundscape

By
ALEXA HENDERSON

SUPERVISORY COMMITTEE:

James Oliverio, CHAIR
Angelos Barmpoutis, CO-CHAIR
Anna Williams, MEMBER

A PROJECT IN LIEU OF THESIS PRESENTED TO THE COLLEGE OF FINE ARTS
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF ARTS

UNIVERSITY OF FLORIDA

2012

ACKNOWLEDGEMENTS

My extreme gratitude to Dr. Angelos Barmpoutis, for his immense help with the logic of Organic Sound and for his knowledge of acoustics. Professor Anna Williams, for her hours helping to teach me the basics of programming, her fine-tuning of the aesthetics of Organic Sound, and for helping me to stay on track with the project. And of course, many thanks to James Oliverio, for believing in me and helping to support me through the entire experience of graduate school at the Digital Worlds Institute.

Summary of Project in Lieu of Thesis
Presented to the College of Fine Arts of the University of Florida
in Partial Fulfillment of the Requirements for the
Degree of Master of Arts

ORGANIC SOUND: A Living Soundscape

By

Alexa Henderson

December 2012

Chair: James Oliverio
Co-Chair: Angelos Barmpoutis
Major: Digital Arts and Sciences

Organic Sound is a music visualizer that expands the expectations of the viewer regarding the forms generated by sound. Attempts by artists in the fine arts, experimental film, contemporary installations, multimedia, and performing arts over the past hundred years to create a visual code for music have championed a genre based on the concepts of synesthesia and musical analogy. The resulting works of art have utilized abstract forms and colors to convey a visual message throughout the annals of visual music.

Organic Sound seeks to push the boundaries between music visualization and motion picture, generating visuals that mimic lifelike representations from our biotic surroundings. Displaying the properties of intensity, pitch and waveform through material associations, Organic Sound will convey these qualities of sound, while creating a visual story for the viewer to consider, contemplate, and enjoy. Microphone input of sound will allow Organic Sound to respond to the viewer/participant's utterances and preferences, thus creating an interactive experience between the participant, the music, and the visual landscapes generated therefrom.

TABLE OF CONTENTS

Introduction

The goal of my music visualizer is to create an application that brings music to life visually in a more realistic way than has previously been experienced. Music visualizers have historically been relegated to the realm of abstraction in their depiction of sound, relying on color, waveforms, shapes and movements to convey the “essence” of a musical score. The concept of synesthesia, or sense perception of one kind translating into the sensory experience of another, has informed the manifestation of visual representations we have come to expect for sound in the realms of the fine arts [1]. In the first American exhibition of its kind, *Visual Music* charts the influences of music and synesthesia on the arts over the past hundred years. In the book based on the exhibition, the authors make the argument that it is from this lexicon that modern and contemporary artists have experimented to “invent a kinetic nonrepresentational art akin to instrumental music” [2].

Organic Sound seeks to break through the barrier of abstraction into something more sensorially engaging, drawing from real life imagery based in our corporeal experiences. Because the focus of this visual application is to mimic our experiential surroundings, it is important that the system’s visible assets resemble, or approximate, their existing counterparts. It is also important that the music visualizer contain visual queues which display key aspects of sound, so that the viewer is able to follow, both visually and auditorily, the different properties of the music being played.

Chapter 1: Motivation

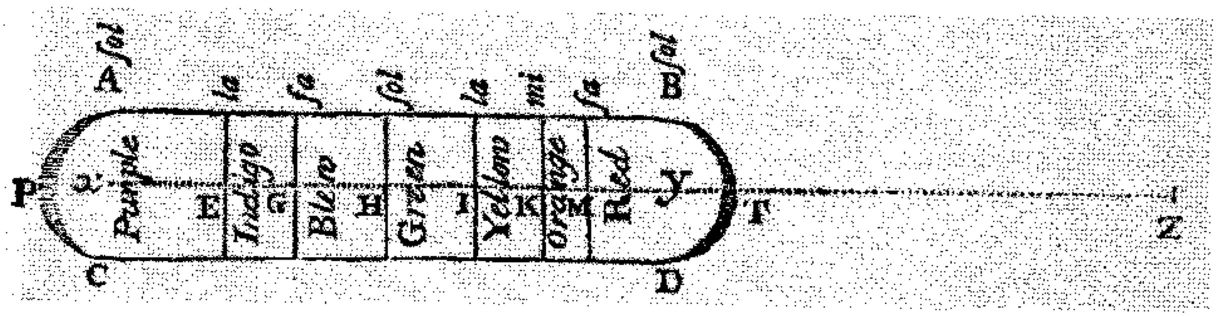
I am interested in creating a music visualizer that engages the viewer in a more complete fashion than what has been attempted in the past. Somewhere between visualizer and cinema, Organic Sound will be built harnessing representations from our experiential surroundings; If the visuals generated from Organic Sound mimic some visceral experience for the viewer, she may be more completely drawn into the narrative of the landscape.

It is also important to me that Organic Sound be interactive in its applications. For this purpose the program will be built harnessing microphone input for sound, allowing for Organic Sound's possible uses in live performance. The direct feedback of the system to the viewer/participant will give the user or musician the capacity to generate the visuals on screen. These graphics, in turn, might inform the viewer/participant as to what pitch, volume, or pattern they might choose to play.

Chapter 2: Historic and Existing Music Visualizers

2.1 Color-Tone Analogy and the History of the Light Organ

Early sound to visual analogies developed out of the idea that a "musurgia universalis" underlay all natural phenomenon. Athanasius Kircher, the 17th century German Jesuit scholar and mathematician who helped to champion this view, based his arguments on observation of the natural world and analogy [3]. Color theory was an area much more easily quantified and though, according to Voltaire, another prominent scholar in the 17th century, Kircher may have been the source of Isaac Newton's interest in studying light and sound, Newton's scientific methodology popularized the color-tone analogy in the 1670’s and helped to fixate further contemplation on musical harmonies and natural phenomenon to the realm of color [4].



2.1.1: Newton’s illustration of the color-tone analogy showing the widths of the seven bands of color in the same harmonic ratios as the notes on a musical scale

The Ocular Harpsichord, the first instrument envisioned for display of the color-tone analogy, was theorized by Louis Bertrand Castel in the 1725 [5]. Though its potential invention caused much debate and speculation as to the validity of perception and the efficacy of scientific inquiry, the Ocular Harpsichord remained an interesting thought experiment and was probably never built during Castel’s lifetime [6]. Other versions of the ocular harpsichord continued to be re-envisioned and built in the decades following Castel’s apotheosized thought experiment, though with less public knowledge and attention.

In the nineteenth century the famous British painter Alexander Wallace Rimington designed an electric color organ. “Music” from his organ became wildly popular in London and Rimington wrote an important book on his achievements, *Color-Music, the Art of Mobile Colour* [7]. In 1911 the Russian composer and pianist, Alexander Scriabin, used Remmington’s Colour Organ for the premiere of his composition “Prometheus: a poem of fire”, one of the few major orchestral works in history to include a color score [8].

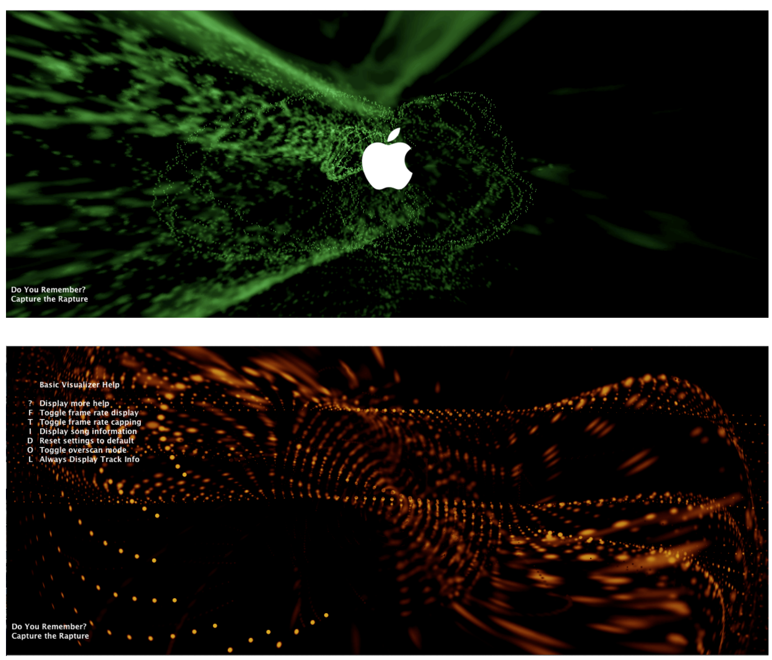
Scriabin also had adamant color to note associations based on Newton's *Opticks*, which ascribed the circle of fifths to the color wheel [9].

Thomas Wilfred's development of the Clavilux in the 1920's, a modern ocular harpsichord, meaning "light played by key", helped to further codify the emerging art form of light, which Wilfred coined "lumia" [10]. With the invention of the Clavilux, color organs had evolved to include three aspects of the visual experience: form, color and motion.

2.1.2: Image generated from Thomas Wilfred's Clavilux

2.2 Contemporary Music Visualizers and Assessments

Contemporary music visualizers have sprung from these color-based, ambient interpretations of music and form. We see parallels as we move from Thomas Wilfred's lumia to the music visualizers currently on the market. Apple's iTunes music visualizers are some of the most widely known. The Classic visualizer released in iTunes in 2001 demonstrates



2.2.1: Screenshots from iTunes Classic music visualizer close resemblances to the aesthetics of abstract form, color and motion seen in Wilfred's art.

The experience of "viewing the music" through iTunes' Classic music visualizer, however, returns tenuous analogies between sound and form. Lines, loosely resembling waveforms, or affected by them, zig-zag, shoot, or swirl across the screen in various colors, widths and meanderings. The visuals are not affected differently by one song or the next. The visuals will never approximate their interpretation of the song twice, and when the music is stopped, the animation continues indefinitely. What we experience is more a kaleidoscope of imagery than visuals driven by the music.

Apple's more updated version of the music visualizer, released in 2010, makes more of an attempt to harness some of the characteristics of music. When the user presses play, several orbs, emitting nebula-like particles, light up or begin exhibiting subsets of programmed characteristics. The animated orbs pulse and change in tempo with the song. The rate of change of the animations seems to be tied to the beats per minute as well. When the user pauses the song, the animated orbs return to some version of their "pre-sonified" form, resembling lens flares (or the absence of light), and solely emit nebula-like particles that fill the screen. Replay of a song does not yield similar visual results, however, and the overall impression is that the animated characteristics of the orbs cycle through pre-programmed permutations. Though this is visually interesting, there is no obvious correlation between the parameter of the sound that created each specific animation.

The updated iTunes music visualizer touts user interaction with the visuals in real time while the viewer watches and listens to the music. Apple offers five main options that allow the user to manipulate the look of the graphics via key commands. Change mode scrolls the visualizer through the orbs' various programmed parameters, generating different visual attributes for the orbs that alters their behaviors slightly. "Change palette mode", which changes the color nebula particles emitted from the orbs, "Toggle freeze mode" alternately suspends and re-animates the orbs at their current state of generation while allows the camera to rock through the scene in a gentle fashion. "Toggle nebula mode" turns on or off the micro particle clouds (nebula) that surround the primary, spawning orbs, and "Toggle camera lock mode" stops the rocking and panning motion that occurs throughout the animation. This version of Apple iTunes' music visualizer is more engaging than the Classic visualizer, but continues in the model of the abstractionist analogy that modern and contemporary art have championed for the vision of sound.

G-Force is a for-payment music visualizer that claims a robust library of visualization features and interactivity. G-Force includes a "line-in" capability for audio, allowing the user to visualize graphics in real-time. The animations generated by G-Force are somewhat akin to those of iTunes' Classic visualizer. Sound is relegated to line-based forms, which "ripple" out across the screen in wavelike associations, changing in color, size, shape and movement. The controls are more extensive than iTunes', and the movement/undulations of the animations are more obviously tied to the beats per minute of the song, like those of iTunes' updated visualizer.

Softskies is another for-payment "music visualizer", created by the same company as G-Force, but which attempts to step into the world of realism. Softskies generates cloudscapes that the song's parameters supposedly effect. The clouds change size, shape, and luminosity as the sky changes colors. The clouds are pleasant, but there is no visual link between the music and the particles emitted, or the colors on the screen. Clouds spawn and change at the same rate when a song is playing as when it is stopped. The tone or loudness of the song do not seem to effect color density of the clouds. Softskies presents as a generative animation with little to no link to sound.

Endless Forest is a music visualizer that greatly influenced the initial idea for Organic Sound. Endless Forest is part of Google's Chrome Experiments, a web-based initiative by Google, to promote innovation and creativity on the web [11]. The application is the only other audio visualizer I have researched that ventures into the world of realism. Endless Forest sets the viewer in an animated, winter-like forest. A song plays while the camera marches through the woods in a continuous forward motion. Snow particles fall lightly or more forcefully in conjunction with the intensity of the music, and an owl flies overhead, against a backdrop of stylized vector trees, a purple night sky and a full moon. The experience is soothing and hypnotic, but limited to one movement, one song, and one stylistic choice. Endless Forest is a step towards realism, but more of a continuous animation set to sound, and fairly limited in its scope.

In my research on the technology available in music visualization, the most cutting-edge audio-visual applications fail to utilize the impact of realistic imagery in their portrayal of sound, or do not develop tone-related analogies to their fullest expressions. Music visualizers that convey obvious correlations between sound and form, using imagery based on the corporeal world, do not exist. Organic Sound will attempt to bridge this gap, correlating acoustic phenomena with its visual counterparts, through the harnessing of realistic imagery.

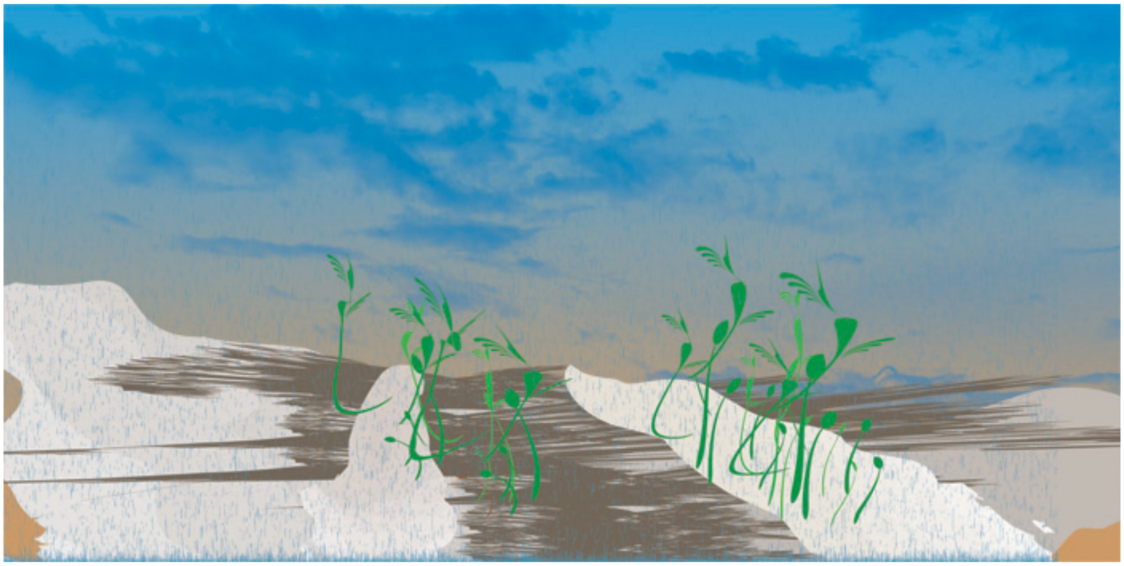
Chapter 3: Challenges

Organic Sound seeks to reveal a narrative to the viewer through the harnessing of specific properties from music played. The correlation of music with the natural order of the planet is not an original idea; it is a scientific and philosophic inquiry personal to great minds of the ages, such as Plato, Kepler, Voltaire and Newton. In the book, "Music Theory and the Natural Order", Daniel K. L, Chua writes:

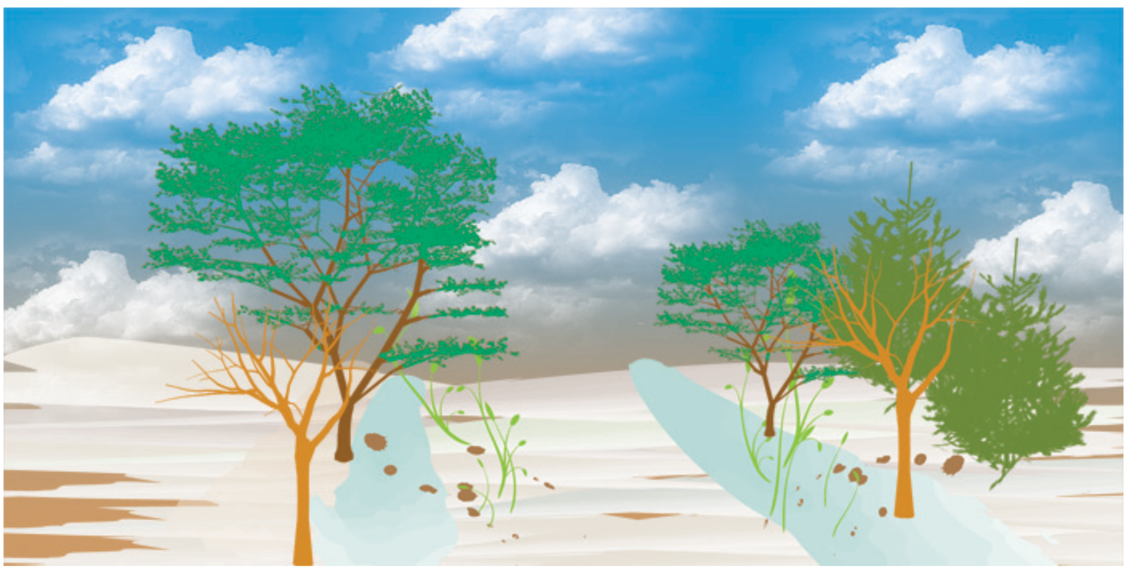
> If anything, the ancient world was far more rational than the modern world in its organization of the cosmos, for it's music was ratio-nality itself; this is why music was not classified as an art but belonged with geometry, astronomy and arithmetic in the sciences of the quavidrium. In Plato's account of creation, music tunes the cosmos according to the Pythagorean ratios, and scales the human soul to the same proportions. This enabled the inaudible sounds of the heavens to vibrate within the earthly soul, and conversely, for the audible tones of human music to reflect the celestial spheres, so that heaven and earth could be harmonised within the unity of a well-tuned scale. This scale came to be pictured as a monochord that connected the stars to earth like a long piece of string that vibrated the structure of the universe [12].

With the idea of using nature as the basis of my music visualizer, I began envisioning a field where recursion trees would "grow" themselves, harnessing specific attributes of the sound to affect tree growth, tree type, ground color, sky color, camera pace, camera path, and to determine whether the sky was clear, stormy or full of clouds.

The proper platform for the generation of Organic Sound was an initial consideration for the project. Pure Data, a free, open source, realtime graphical programming environment for audio, video and graphics, is a program used successfully in the generation of music-driven graphics [13]. Though the programming



3.1: initial sketch of the development of Organic Sound, Oct 2011
“Lower frequencies and a slower pace of the music might generate darker colors and perhaps the element of rain”



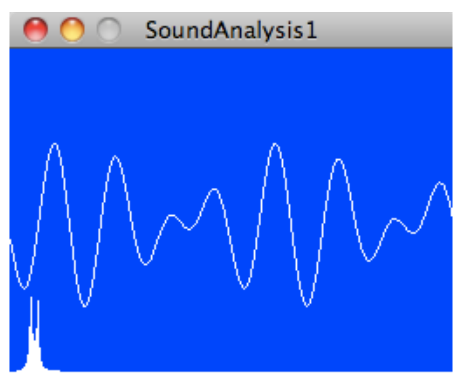
3.2 : initial sketch of the development of Organic Sound, Oct 2011
“An upbeat tempo and major key might correspond to a generally sunny and verdant forest”

modules within Pure Data could facilitate the musical analogy of specified characteristics of sound driving visual design choices by the user, the capacity of Pure Data to generate realistic recursion trees, initially a main design component of Organic Sound, was not evident.

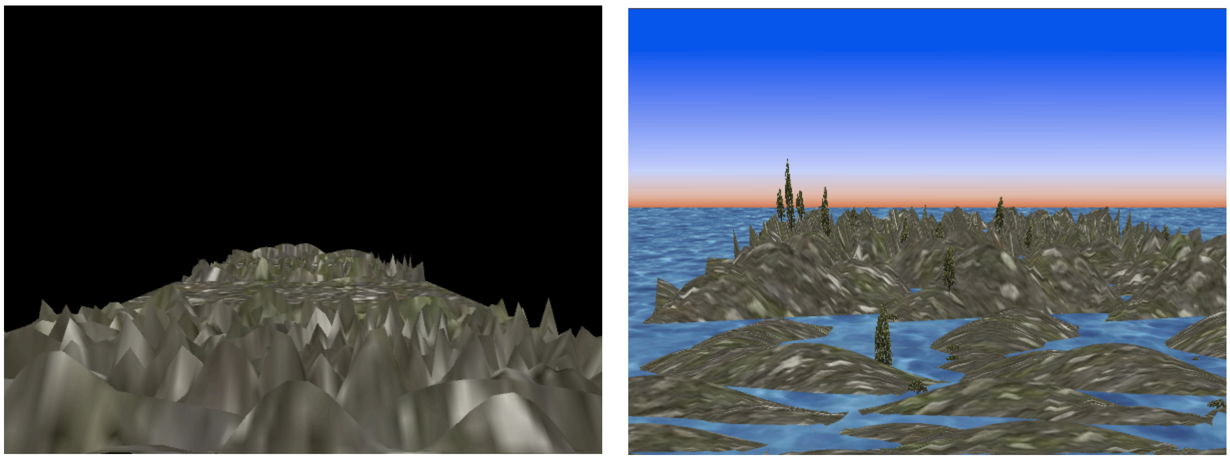
My experience with Max/MSP, a proprietary graphical programming application, initially developed by the same inventor as Pure Data, led to a similar conclusion [14]. Both programs, adequate for the analogy-based music visualizers envisioned in the vein of the contemporary fine arts, would not lend themselves well to realistic terrains and tree growth. Traditional forms of code and logic would need to be utilized for the task of generating realism in virtual space and the objective of the viewer experiencing verdant life unfolding around them.

Processing, another free, open-source platform for programming visuals, music and video is the program I chose for building Open Sound. Processing utilizes a stripped-down syntax, based on java, which is easy for beginning programmers and artists to understand and build their first applications with [15]. Though the syntax is much more simple than most programming languages, the logic used in Processing is universal programming; the program would allow for the complex generation of dynamically generated, 3D trees, utilizing if/else statements and for loops in the code.

Another advantage of using Processing as the program with which to build Organic Sound is the ability to plug in third party libraries that add functionality to Processing's framework. Organic Sound requires the use of a library that will process sound and divide and analyze its components. A Library developed by Krister Olson, "Tree Axis", provided the capacities necessary for the application. The first patch I utilized for the analysis of sound used mp3 import while displaying an



3.3: Original version of Krister Olson's "Analysis" patch, with the oscilloscope at center and spectral analysis at the bottom left oscilloscope and performing real-time spectrum analysis (see Fig 5). With the help of Dr. Barmpoutis, I used this sound analysis functionality to build the first versions of Organic Sound, seen below.



3.4 : First patches with oscilloscope and spectrum analysis of mp3 files

These beginning patches build the terrain using a 2D array. Every 512 samples of sound are converted into a waveform which displays visually the amount of pressure exhibited by the samples over time. The wavelength and the speed of the wave determine the pitch of the sound. The longer the wavelength (one full cycle from the zero point of the wave traveling up and the zero point traveling down), the lower the pitch. The amplitude of the wave represents how loud the sound samples are [16].

For these first arrays, the data for each 512 samples was divided into 50 positions on the X-axis. Each subsequent set of samples is moved forward and stored in the second slot of the 2D array and thus the effect is that the terrain is running away from the camera for 100 positions into the Z-space of the patch. A PNG image of a tree was then added to the patch to convey the data processed from the spectrum analysis. I will explain in more detail later how the spectrum analysis operates, but for now it is sufficient to say that the various heights of the trees represent the amounts of sound at each pitch. This sound (represented by the tree) is then divided randomly across the X-axis to mimic something more organic than what the frequency of each pitch yields in the original spectral analysis (See Figure 4.1).

Initially, I had hoped to traverse through X, Y and Z-space in the patch, and have lifelike trees and fauna grow up around the viewer/listener. This original envisioning of Organic Sound would give the user ample time to enjoy the unfolding of the experience, as the pace of the camera movement would have originally been tied to beats per minute. The original pace of the application would necessitate that the trees look realistic or stylistically compelling and would also necessitate that the trees be able to "grow" themselves. For both the movement through Z-space and the realistic experience of a growing forest, Dr. Barmpoutis suggested that we implement Open-GL into the application, to allow for the rendering of 3D objects and 3D space in the patch.

3.5 3D, dynamically generated trees with randomized growth variability

Open-GL, or "Open Graphics Library", is the industry standard for developing interactive 2D and 3D graphics applications [17]. It is multi-language, cross platform, open-source application programming interface (API) which interacts with the computer's graphics processing unit (GPU) to achieve hardware-accelerated rendering, helpful in the rendering of graphics-heavy programs [18]. Dr. Barmpoutis' library, "UFDWlib", was utilized as the foundation for Open-GL in the Organic Sound. Adding this functionality into the code requires the use of more complicated syntax in the application and also changes the coordinates for placing objects in the patch, which can lead to confusion and added layers of troubleshooting when components are added or are not working properly.

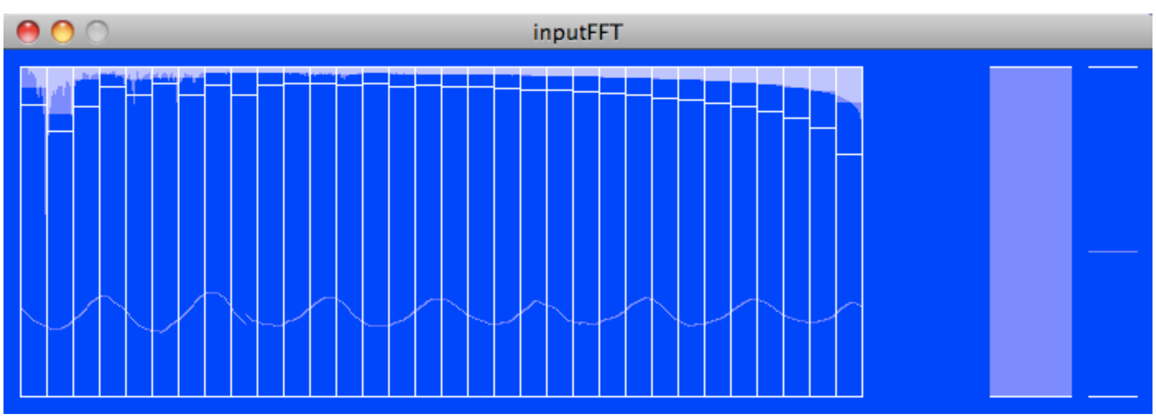
The initial error in my logic for the building of Organic Sound was that any computer program could both process realtime inputs of data from sound for the growth of the forest, as well as allow the camera to meander through the virtual landscape based on some property of the music, at the same time. This is a logical impossibility as realtime processing of data necessitates a one-to-one correlation between sound processed and visuals displayed.

I was faced with the decision to choose between flexibility of virtual movement through space using pre-processed data from pre-recorded music, or the option of interaction with the system in realtime, allowing for its potential use in live performance mode and the viewer's possible participation with the visuals generated. If I did not want to use a library that would process the data of a song in it's entirety, allowing for more flexibility in the unfolding of the visual experience during playback of a song, what became one of the main challenges in the development of Organic Sound was: How do I convey an unfolding event, clearly linked to the music through visual forms, that can be experienced in the time it takes for one hundred sets of music samples to be processed by the application?

CHAPTER 4: RESULTS

The code I prepared previous to this new understanding of logic and the more complete comprehension of the parameters inherent therein would require some reworking. The pace of the application, necessitated by the realtime analysis of sound, forced me to rethink the tree growth and generation that Organic Sound would display. The 3D generation of assets requires heavy processing by the graphics processing unit; without the capacity to slow down the visual plane and experience the process of the trees expanding, there would be no reason to keep such functionality in the patch.

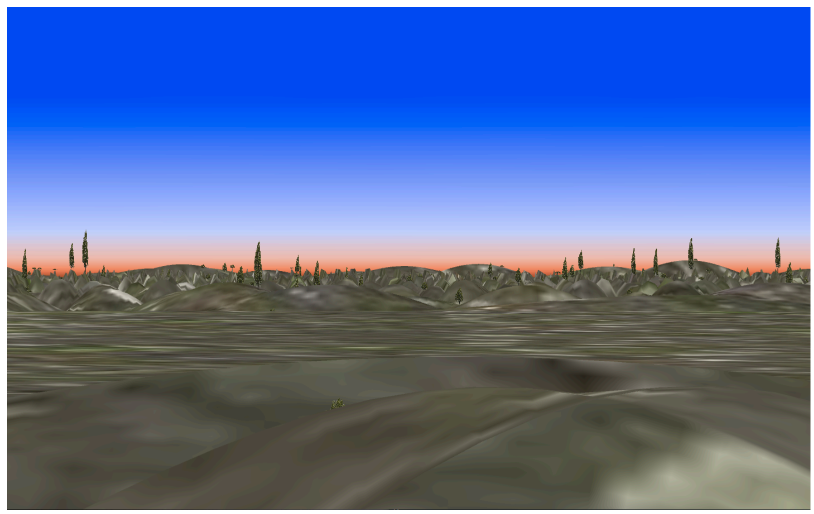
Portions of Organic Sound already built using Krister Olson's "Analysis" code (shown previously in figure 5), process pre-recorded mp3 files. In order to achieve the microphone line in functionality I desired, I would need to adapt another of Olson's Tree-Axis patches, modified from the code of Marius Watz, "InputFFT". InputFFT, similarly to Analysis, draws the waveform of the processed samples, mimicking an oscilloscope, and analyses the frequency, or amount of each pitch, comprising each sample.



4.1: Olson's original InputFFT analysis, for microphone in functionality

Using the new patch from Krister Olson's library, I sectioned off the InputFFT code into its own class and fused the properties of the previous SoundAnalysis with the functionality of the new microphone line in capacity. The main difference between the previous code in the new line in code is that SoundAnalysis calls for an audio channel to be opened in its constructor whereas InputFFT calls for an audio input. Similarly, to call the audio channel, the user must declare the filename of the mp3 for SoundAnalysis, whereas the audio input takes in sound samples of a specific size. Both classes analyze and display the sample's waveforms and calculate the frequency spectra.

In the updated version of the music visualizer, the new InputFFT data populates the wavesample and frequency arrays with which Organic Sound draws the ground and generates the trees and tree heights from. For the purposes of drawing the viewer *into* the action of the scene, the 2D array needed to be modified to draw the wavesamples and frequency data in the last position of the Z-space slot in the array, and with each subsequent frame, translate the data forward one position *towards* the viewer (instead of away, as previously coded). I also doubled the size of the array in both the X and Z directions, so that the terrain stretches from one side of the canvas to the other, and extends far enough out to achieve further depth perspective.



4.2: InputFFT class tied to the terrain and tree generation in Organic Sound

An important and additional parameter that the InputFFT class provides the application is the calculation of intensity, or volume. With this additional data from the song, Organic Sound now has three sets of data with which to draw the landscape: waveform, frequency and intensity. The components of sound are tied to the landscape in the following ways: wavesamples draw the topography of the sketch, frequencies select which types of trees to draw and determines their heights. The intensity, or volume of the sound, is tied to the color of the ground, and also controls the darkness of the sky.

Olson's initial InputFFT graph divides the 256 sets of frequency spectra data into 32 bands and averages their sums. When I tested the range of detection for the InputFFT patch, beginning at 20Hz and ranging up to 20,000Hz, the first 24 bands of the graph displayed with obvious correlation to the sound. The most obvious way to select for the types of trees displayed by Organic Sound is to choose a bush or tree for the first 24 working averages in the FFT and display each tree for the frequencies analyzed, according to the amount they are detected in that sample.

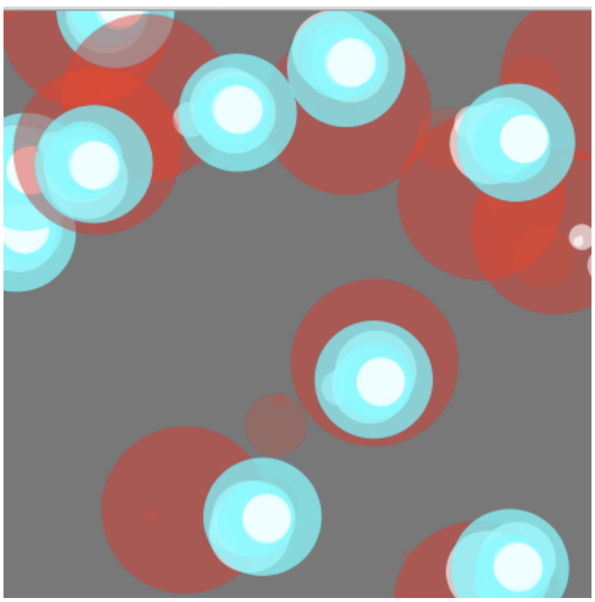
A visual analogy could also be to link light colored, nascent trees to higher pitches and darker, fuller, and/or older trees to lower pitches. Initially, the choice had been made to randomize the distribution of frequencies, via the heights of trees, across every row of 512 music samples. The result was perhaps a more natural looking terrain, but as more tree types were introduced to the patch, less analogy was visible between

4.3: Trees chosen for Organic Sound linking high frequencies to light, young trees versus more mature, fuller trees for low frequencies the frequencies processed and the trees, so the randomization of the tree heights along the X-axis was removed from Organic Sound.

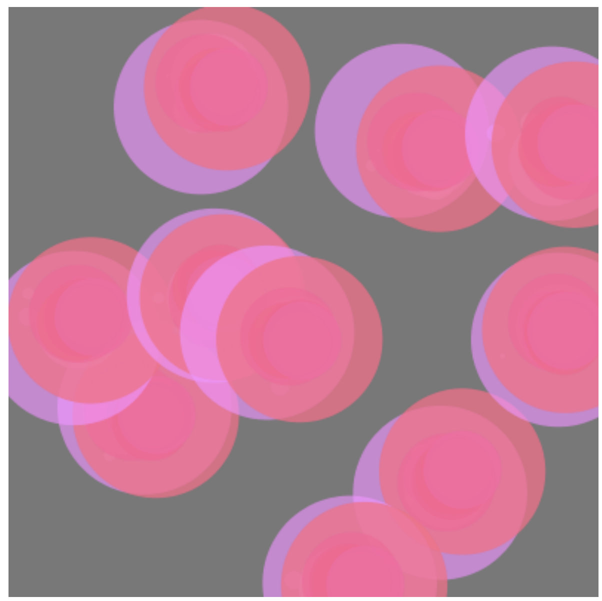
In this most recent iteration of Organic Sound, the intensity of the samples effect the hue of the ground texture. When a threshold of intensity is hit by the level of the sound being interpreted, the color of the ground begins to change and become more intense. The hue of the tree textures are affected equally by this parameter, and the result is a dynamic visual change that further links the music to the visual landscape. Using a related analogy, the sky in Organic Sound lights up according to the intensity of the sound. Thus, quiet music yields a dusk-like experience of the terrain. Once a threshold is reached by the sound level, the sky remains illuminated until the end of the song, when the absence of sound returns the sky back to its initial state of darkness.

4.4: Intensity of sound yields more vibrant hues in the terrain & illuminates the sky from its initial state of darkness

Historic musical analogies correlated color with pitch. Thus far in my visual application, the only reference to frequency was to display the heights of the trees based on the amount of every spectral band contained in each music sample processed by the FFT analysis. In an effort to incorporate elements of the enthusiastic color-tone analogy into Organic Sound, I added particles into the patch to mimic clouds in the sky. These particles change size based on the volume of the music and color based on the pitch detected at the time of their emittance: lower volumes yield smaller particles than high volumes and lower frequencies yield darker toned particles than do higher frequencies.

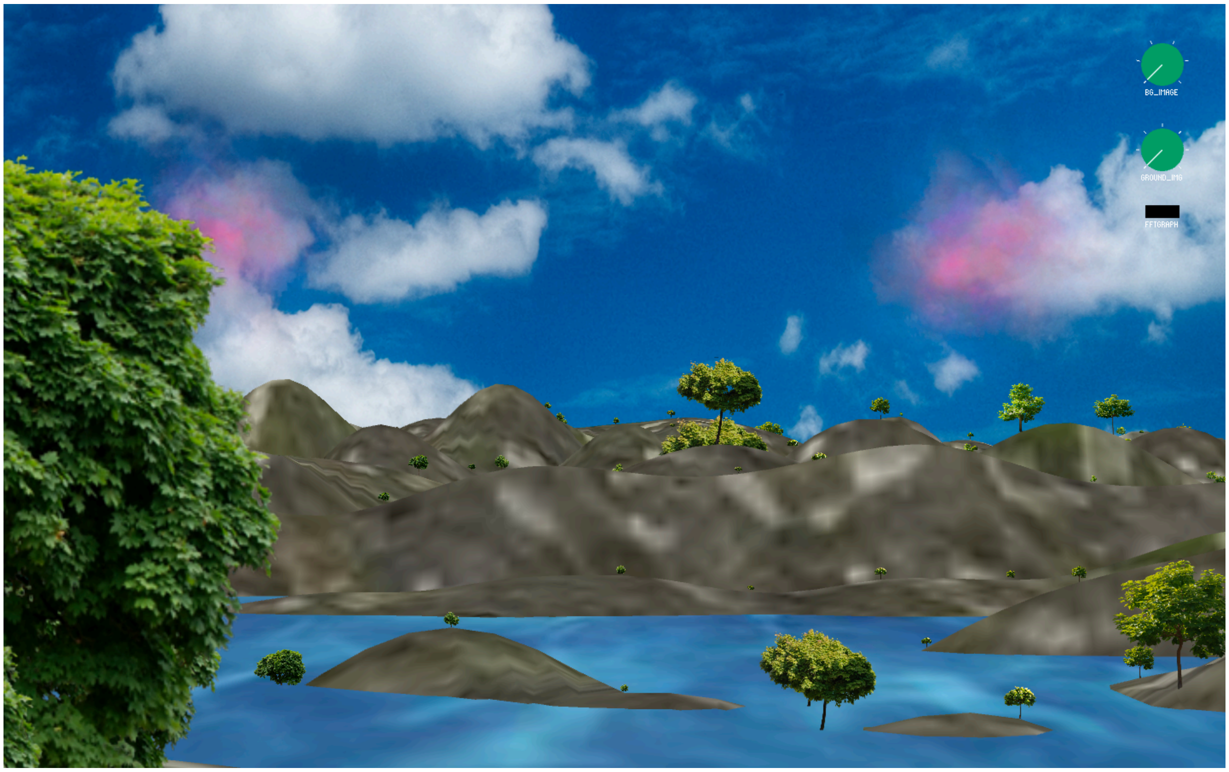


4.5: Transition from lower-pitched, louder music to higher-pitched softer volumes



4.6: Mid-level frequencies played at medium volume

To finalize the realistic effect of clouds in the sky, the code was reworked to add a png image of a cloud wisp instead of the abstract ellipses shown above. Randomized rotation was added to each png as it it emitted, to give a more cloud-like aesthetic, and the emission rate and colors of the particles were adjusted, as the colors reacted differently with white backgrounds than they had initially. The effect are clouds that approximate realistic versions and fit more naturally with the entirety of the system. The decision was also made to add a flat plane to each ground terrain, to mimic either water or an alternate ground covering. This allows more visual contrast within Organic Sound and makes the amplitudes of the wavesamples more obvious against a flat, contrasting surface.



4.7: Realistic clouds add to the natural aesthetic of Organic Sound. The addition of a flat plane into the scene adds to visual breakup of the space

One last feature of the visuals in Organic Sound are tied to the sound: camera height. A destination height, or vantage point from which the user will watch Organic Sound unfold from, based on the heights of the upcoming wavesamples, was assigned to the camera. For every 20 frames, the system adds together the height data for a specific slot (ground placement) in the 2D array and then divides by a relatively low number, giving the higher numbers more dominance. The difference between the camera's current height and this optimal "destination" height is narrowed in on with each frame, but constrained to a minute increment of change, so that the camera will clear the topology of the terrain as it passes under the camera, but will not appear to be jostled.

Originally assuming that Organic Sound would process a piece of music and analyze its complex attributes, such as harmony/discord or key, the initial idea was that data would purely and completely drive Organic Sound's choice of terrain, tree type, sky background, colors, etc. The modification of Organic Sound to an application with realtime capacity limited the scope of musical analysis possible, and necessitated the addition of user-inputs to manipulate the visuals generated by the system.

"ControlP5", a third-party library for Processing, was added into Organic Sound for this purpose. Knobs and buttons extend the expressive capacity of the system to generate different backgrounds for the sky and different textures for the ground and trees. The result is a music visualizer that matches an extensive array of variability in musical tones and timbres with its visual codes, and which can be enjoyed as both a personal and dynamic experience, not unlike sound itself.

CITATIONS

1 Brougher, Strick, Wiseman, and Zudith Zilczer. *Visual music : synaesthesia in art and music since 1900* (California : Museum of Contemorary Art, 2005) 14.

2 Brougher, 7.

3 Hankins and Robert J. Silverman. *Instruments and the Imagination*. (New Jersey : Princeton University Press, 1995) 76.

4 Hankins, 75.

5 Hankins, 73.

6 Hankins, 76.

7 Jones, Tom Douglas. *The Art of Light and Color; featuring mobile color expression, lumia, kinetic light--with instructions for the creation of dramatic color and light instruments*. (New York : Van Nostrand Reinhold, 1972) 20.

8 Hankins, 85

9 "Alexander Scriabin." *Wikipedia, the Free Encyclopedia*. Web. 31 Oct. 2012. <http://en.wikipedia.org/wiki/Alexander_Scriabin>.

10 "Thomas Wilfred". *Wikipedia, the Free Encyclopedia*. Web. 2 Nov. 2012. <http://en.wikipedia.org/wiki/Thomas_Wilfred>.

11 Endless Forest, web

12 Natural Order, 22

13 Pure Data, web

14 Max Wiki, web

15 Processing, web

16 Hollis, web

17 OpenGL site

18 OpenGL wikipedia

BIOGRAPHICAL SKETCH

Alexa Rose Henderson was born in Gainesville, Florida, in 1981. She attended the International Baccalaureate program at Eastside High for her first two years of high school, followed by a year at Interlochen Arts Academy in Interlochen, Michigan, for acting, and a final year at Santa Fe College, as a dual enrollment student. Alexa's career in design began at Santa Fe Community College, where, after receiving an A.A. in general studies, she went on to acquire an A.S. degree in Graphic Design Technologies. Following the completion of this degree in 2003, Alexa worked as a web designer and developer at 352 Media Group and went on to do freelance graphic design and web development.

After a stint in New York City in 2006, Alexa moved back to Gainesville, Florida, to finish her Bachelor's Degree at the University of Florida in 2009, where she studied English (Film and Media Studies) and minored in Sustainable Studies. Alexa worked at the Digital Worlds Institute as a secretary and web developer during her final year of undergraduate schooling and applied for candidacy in the Digital Worlds Institute's Masters of Arts in Digital Arts in Sciences degree upon graduating with her B.A.

During her time in the M.A. in D.A.S., Alexa focused on projection design and music visualization. Her thesis work is the creation of a new style of music visualizer, utilizing cinematic realism to portray attributes of music. Upon completion of her M.A., Alexa Henderson is seeking employment with production companies, specializing in projection and design for events, and pursuing rock concerts and music festivals for work as a freelance projection designer and music visualization specialist.