GENERATING AUDIO-RESPONSIVE VIDEO IMAGES IN REAL-TIME FOR A LIVE SYMPHONY PERFORMANCE

A Thesis

by

ALLISON BROOKE BEANE

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

May 2007

Major Subject: Visualization Sciences

GENERATING AUDIO-RESPONSIVE VIDEO IMAGES IN REAL-TIME FOR A LIVE SYMPHONY PERFORMANCE

A Thesis

by

ALLISON BROOKE BEANE

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Approved by:

Chair of Committee,	Karen Hillier
Committee Members,	Carol LaFayette
	Jeffrey Morris
Head of Department,	Mark Clayton

May 2007

Major Subject: Visualization Sciences

ABSTRACT

Generating Audio-Responsive Video Images in Real-Time
for a Live Symphony Performance. (May 2007)
Allison Brooke Beane, B.E.D., Texas A&M University
Chair of Advisory Committee: Prof. Karen Hillier

Multimedia performances, uniting music and interactive images, are a unique form of entertainment that has been explored by artists for centuries. This audio-visual combination has evolved from rudimentary devices generating visuals for single instruments to cutting-edge video image productions for musical groups of all sizes. Throughout this evolution, a common goal has been to create real-time, audio-responsive visuals that accentuate the sound and enhance the performance. This paper explains the creation of a project that produces real-time, audio-responsive and artist interactive visuals to accompany a live musical performance by a symphony orchestra.

On April 23, 2006, this project was performed live with the Brazos Valley Symphony Orchestra. The artist, onstage during the performance, controlled the visual presentation through a user interactive, custom computer program. Using the power of current visualization technology, this digital program was written to manipulate and synchronize images to a musical work. This program uses pre-processed video footage chosen to reflect the energy of the music. The integration of the video imagery into the program became a reiterative testing process that allowed for important adjustments throughout the visual creation process. Other artists are encouraged to use this as a guideline for creating their own audio-visual projects exploring the union of visuals and music.

To my family

ACKNOWLEDGEMENTS

I would like to sincerely thank Professor Karen Hillier, my thesis chair, for her unwavering enthusiasm, guidance and encouragement during this process. Thank you for having faith in me to accomplish such a grand task, and always encouraging me to push my abilities to the next level. Her inspiring words have given me a deeper confidence in my work and my abilities to achieve any task I choose to undertake. I would like to thank Dr. Marcelo Bussiki, Director of the Brazos Valley Symphony, for allowing me to be a part of such an incredible experience. This project would have never come to a realization without his enthusiasm and desire to explore new endeavors; I am greatly appreciative. Thank you Professor Carol LaFayette, my committee member, for sharing her knowledge of the software, and guiding me through the technical and aesthetic process. I am also thankful for the support and encouragement of both Dr. Lee Couch III and Professor Jeff Morris, my other committee members.

I want to thank the Brazos Valley Symphony Orchestra board members for generously donating the initial money to fund this process. I also want to thank Texas A&M University's Academy for the Visual and Performing Arts for supporting my work and granting me funding to complete the project. These generous contributions allowed me to explore all creative possibilities to produce a successful show.

The support of the entire Visualization Laboratory has been greatly appreciated. I could not have made it through this project without the support of my good friends, Tamara Cuellar, Kurt Phillips, Tony Piedra and Katie Van Maanen. Their ability to keep me sane throughout this whole process was impressive and greatly appreciated.

To my loving, supportive, generous and always encouraging family, I thank you from the bottom of my heart. I am blessed to have such a wonderful family who encourages me to strive for excellence, yet allows me to discover my own path.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURES Page

CHAPTER I

INTRODUCTION

I.1. Combining Visuals with a Musical Performance

Visual art and music often share a unique relationship in which one influences the other. Artists, scientists, composers and many others throughout the centuries have explored this idea. Leonardo da Vinci sketched his views of music and its relationship to color during the fifteenth and sixteenth centuries. The artist Wassily Kandinsky intended for his 20th century painting *Green Sound* to be “heard” by the viewer knowing this could not be fully explained in words [11]. From early mechanical devices, which produced images for sound through the use of a pulley system, to the modern interactive technology of VJs (video jockeys) that mix and process video during live performances, the attempt to produce expressive translations between sound and imagery have been frequently explored as a form of entertainment and personal expression. Computers and visualization technology have enabled artists to write their own programs to generate real-time interactive visualizations, which encompass the essence of the music while heightening the performance experience [1, 13]. The concert *Adventure in Sound* incorporated cutting-edge computer visualization technology to generate real-time audio-responsive imagery for a live musical performance by the Brazos Valley Symphony Orchestra. This thesis project was an integral part of the *Adventure in Sound* production, which was performed on April 23, 2006. The integration of visuals into this musical performance added a deeper level of entertainment and expression of the music for the audience.

This thesis follows the style of *IEEE Transactions on Visualization and Computer Graphics*.

Adding visuals to a live musical performance creates stimulation of both the aural and visual senses. By allowing a person to “see” the musical structure, the visual stimulation can affect the impact of the music and cause a deeper appreciation and understanding for the work [11]. The artist can influence the tone and mood of the music with the visuals and vice versa. For example, a piece of music that is based on the eerie whistles, sirens, and booms of space, as in the performance *Sun Rings,* may be supported by visuals that are equally eerie to complement the mood of the piece. NASA images captured in space were used as the backdrop to this performance. If the artist, Willie Williams, had shown images of happy, bouncing children and puppies, the mood conveyed would have been completely different and not as successful. Williams stated that the visuals he chose were sometimes “specific to the source sounds . . . but more often [were] an abstraction based more loosely on the mood of the composition as a whole” [8]. Unifying visuals with a musical performance is therefore important in allowing for complete expression of the intended mood and experience.

Real-time audio-visual interaction allows for an ideal combination of image and sound, making synchronization essential.

I.2. The Need for Audio-Visual Synchronization

Synchronizing the visuals to the music helps the audience understand the intended correlation between music and image. One method for insuring accurate timing is to make the visuals respond to the music in real time when a threshold is reached in a given parameter measured from the music. For instance, when these visuals were performed with the Brazos Valley Symphony Orchestra, a flute’s pitch was being monitored. Each time the sound crossed a predefined threshold, it triggered a filter to affect the video, which created an obvious correlation between that sound and the image. Techniques to produce this type of real-time audio-visual synchronization are explored by artists worldwide. For example, artist Golan Levin is well known for his creation of the AVES system, which allows a user to interactively create undulating, responsive sounds and imagery through gestural mark-making [6]. Real-time audio-visual interaction allows for a perfect combination of image and sound, and compensates for any unexpected change in the musical performance.

I.3. Using a Real-Time Visualization Program

There are several programs available that allow for interactive real-time visualization. While some artists use commercial software, there are advantages to writing custom programs. Max/MSP is "a graphical programming environment for music, audio and multimedia" [2], allowing the artist to create interactive audio programs. Jitter is a component that "supports real-time manipulation of video, 3D graphics and other data sets within a unified processing architecture" [2]. Together, Max/MSP and Jitter allow the artist to freely produce accurately responsive and stunning visuals for a musical performance.

The purpose of this thesis is to combine images with music to enhance the mood and tone of a live symphony orchestra performance. This method was executed through the synchronization of visuals with the music in real time for precision timing, and through the writing of a specialized program to generate an intriguing and responsive visual presentation. This blend of complementary visuals and music creates a unique live performance experience for the audience and participants. Writing a program that generates real-time audio-responsive visuals is important to accomplishing this experience.

I.4. Artistic Statement

This project creates computer generated, audio-responsive visuals that are interactively controlled by the artist and performed with a live symphony orchestra. Heavily processed video images reflect the energy of the music and accent the various tones, musical colors and subtleties within John Adam's *Short Ride in a Fast Machine*. A computer program has been written that is flexible enough to control any synchronization variables and efficiently process multiple video files, producing real-time, interactive results. This project was performed in conjunction with the Brazos Valley Symphony Orchestra on April 23, 2006 in the production titled, *Adventure in Sound*. During the live performance, these images were projected onto a large projection screen directly above the orchestra, creating one visual fabric of responsive image and musicians on stage.

CHAPTER II

PRIOR WORK

II.1. Early Pioneers

Early 18th century artists first attempted to unite sound and images through mechanical devices, igniting the desire to explore this relationship. This interest in audio-visual fusion compelled artists to use the resources available at the time. The first artists used rudimentary devices that generated simplistic imagery such as candlelight and shadows. As technology advanced, so did the possibilities of synthetic art and its integration with music. Today, artists primarily use digital technology to produce complex images.

One of the earliest known devices was created by Father Louis-Bertrand Castel in 1734. His *clavecin oculaire* (ocular harpsichord), combined a traditional harpsichord and a six-foot square frame to display sixty panes of colored glass. The colors chosen were believed by Castel to correspond to the notes of the Western musical scale. He sought "to give the colours, irrespective of their harmonic order, a kind of intensified quality of liveliness and lightness which they inevitably lack upon a canvas without life or motion" [6]. The ocular harpsichord worked by exposing the colored glass panes, each were hidden by a shutter door, when a corresponding key was pressed. By use of a pulley system, the door would open, revealing a glowing colored window illuminated by candlelight [12]. In 1919, Thomas Wilfred designed a "color organ" by using pure light as the art form so that sound and music were either completely connected or disconnected as an accessory to the performance. The Clavilux consisted of a large keyboard with five rows of sliding keys and stops that could be combined to obtain the colors sent to a series of projectors and reflectors

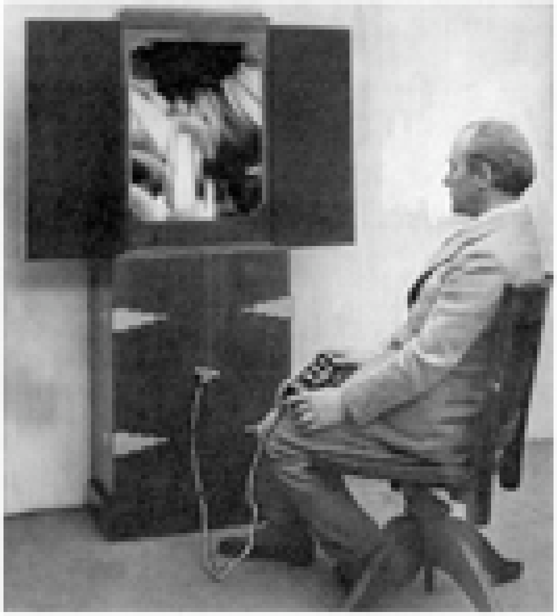


Fig. 1. Thomas Wilfred, “The Clavilux Jr.” 1930 [6]

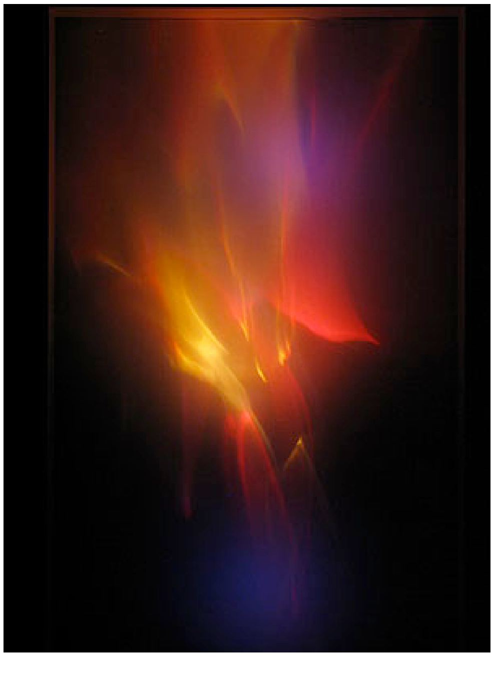


Fig. 2. Thomas Wilfred, Still from “Untitled”, Opus 161, 1965/66 [4]

used to display the image (see Figure 1) [18]. The resulting color projections, called *Lumia* (see Figure 2), "which emphasized the use of slowly metamorphosing, polymorphous streams of fluid color, stand as the earliest surviving color music about which we can make fair aesthetic judgements" [6, 17]. Oskar Fischinger, in the mid 1950's, created the Lumigraph (see Figure 3), which was an interactive system that used people and their motions to generate the visual images in real time [10]. It consisted of a box with an unlit rubber screen covering the front. When a person pushed the rubber outward, the screen would pass through a path of colored light, exposing the rubber and generating color visuals based on the performer's actions. The performance required two people to play the light and was choreographed to music. Each of these artists promoted an interaction between the user, the imagery and the sound [5].

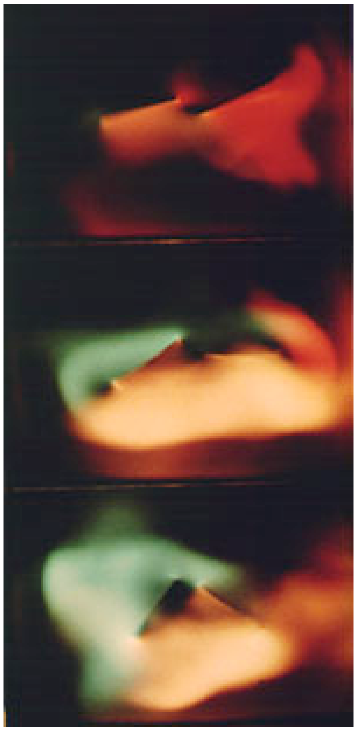


Fig. 3. Oskar Fischinger, "Lumigraph" 1955 [5]

II.2. Modern Audio-Visual Artists

Today, with the use of computers and visualization technology, this idea of visual music has been intensely explored by artists that push the boundaries to generate digitally modern versions of real-time interaction between sound and imagery. Robert Rowe and Eric L. Singer created two systems that manipulate animations based on the real-time analysis of music. The first system, Flock of Words, is a flocking simulation that uses musical input to guide a flock of words and control various attribute. The analyzing software controls the selection of video clips used, the selection of word sets used, the color of the words and background screen, and twelve flocking parameters. The video was projected onto holographic material to create a "diffractive screen for the animation" [14]. Responsive

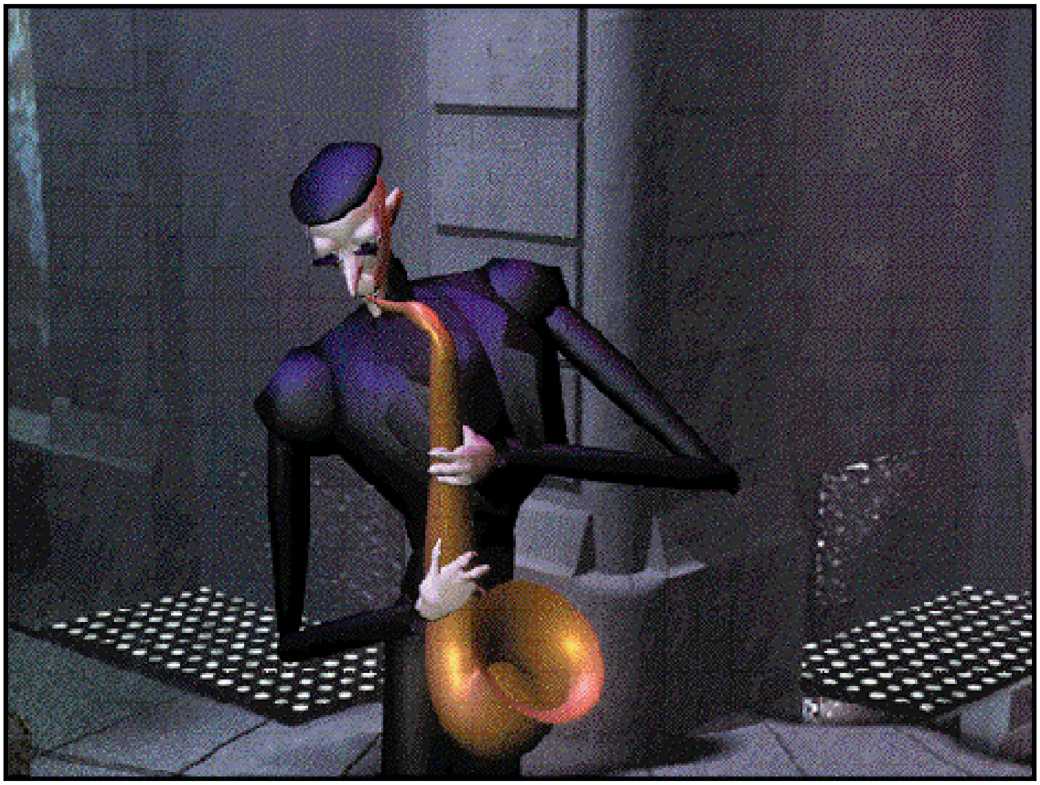


Fig. 4. Robert Rowe and Eric L. Singer, "The Interactive Virtual Musician" [15]

changes in lighting causes the holographic imagery to cross and turn in conjunction with the music. The second system is the Interactive Virtual Musicians system (see Figure 4). It has the option to use Musical Instrument Digital Interface (MIDI) sources such as keyboards, electronic drums, and pitch-to-MIDI converters as sources for manipulation. These inputs are analyzed in real time and used to directly influence the performance of virtual musicians.

Another artist, Golan Levin, created a network of five interactive systems, named Audiovisual Environment Suite (AVES), which made possible the simultaneous performance of animated image and sound. In 2000, while a student at MIT, he used the computer to synthesize graphics and sound in response to real-time gestural inputs. Each system is built to generate an inexhaustible audiovisual substance that allows the user to yield interesting, infinitely variable and personally expressive visual and aural performances. The five systems, Yellowtail, Loom, Warbo, Aurora and Floo, require the user to create and manipulate imagery with the mouse, which then generates sound. The system Yellowtail

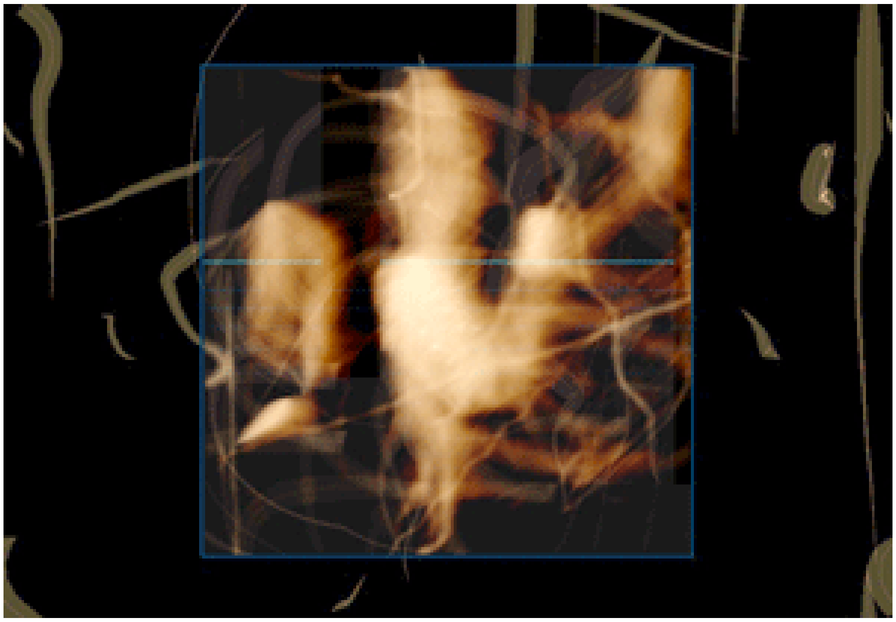


Fig. 5. Golan Levin, Still image from “Yellowtail,” 2000 [6]

(see Figure 5) is a form of animation-generation that repeats a user's stroke end-over-end enabling simultaneous specification of a line's shape and quality of movement. Loom is designed as a reaction to Yellowtail. The user draws lines and marks with the mouse and corresponding audio is produced. The audio is based

on the curvature of the line, pressure, and velocity of the stroke (see Figure 6). The third system, Warbo, allows the user to create animated compositions of glowing "blobs" (see Figure 7). When a second cursor is played over the surface of the spot, it produces tones and chords. The tone gets louder as the cursor approaches the center of the spot. With the Aurora system (see Figure 8), the user creates and manipulates a shimmering, nebulous cloud of color and sound. It rapidly evolves, dissolves and disperses as it follows and responds to the user's movements. The last system, Floo (see Figure 9), disperses and reflects soft-edged tendrils in response to the user's movements, also generating corresponding audio.

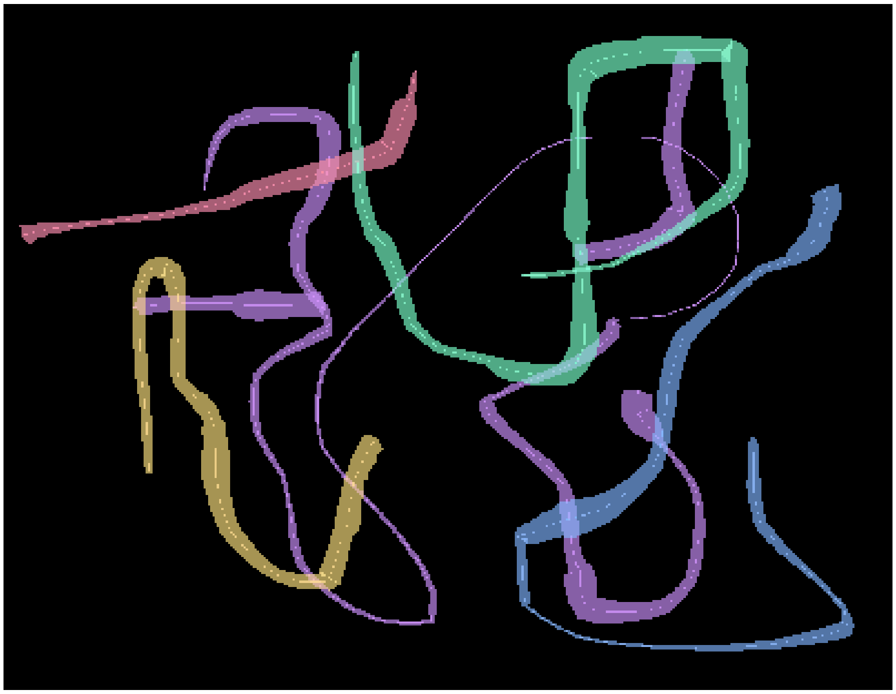


Fig. 6. Golan Levin, Still image from “Loom,” 2000 [6]

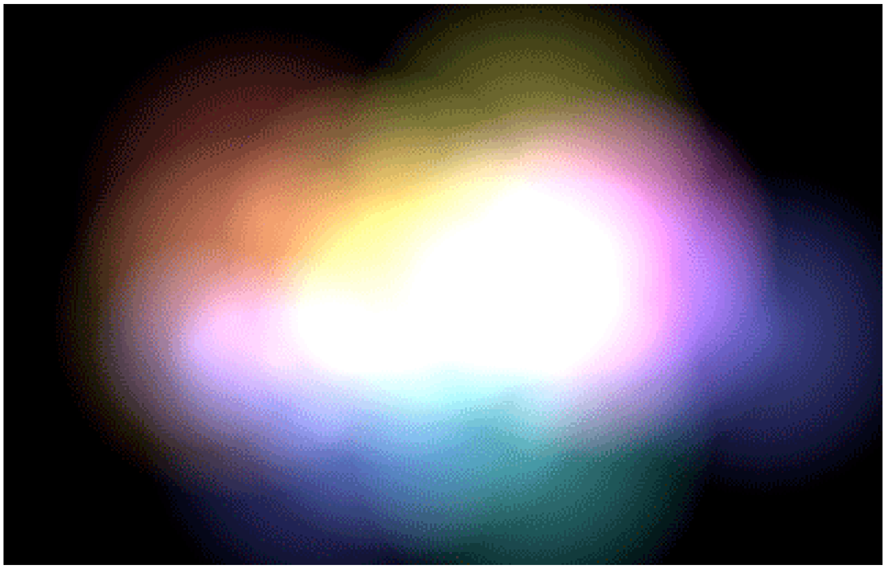


Fig. 7. Golan Levin, Still image from “Warbo,” 2000 [6]

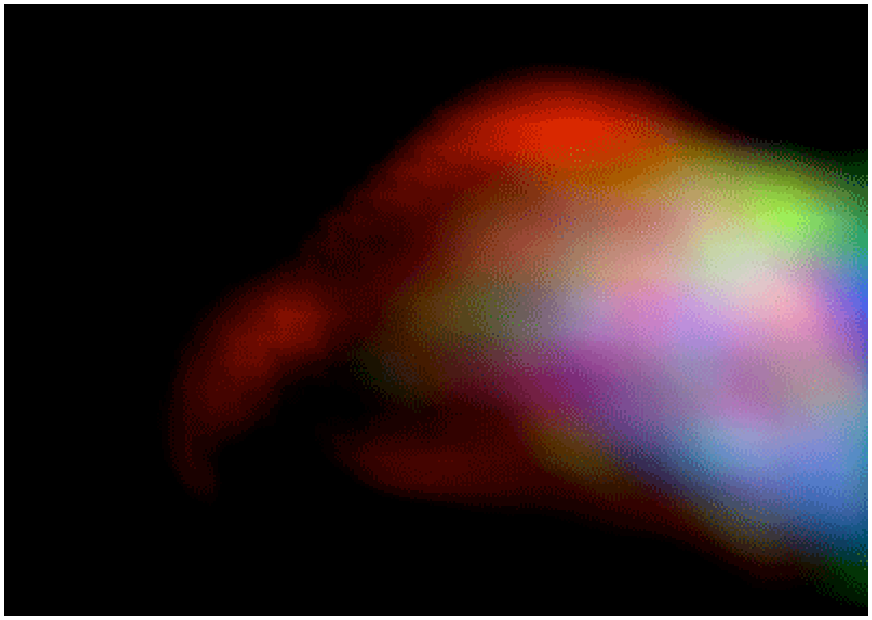


Fig. 8. Golan Levin, Still image from “Aurora,” 2000 [6]

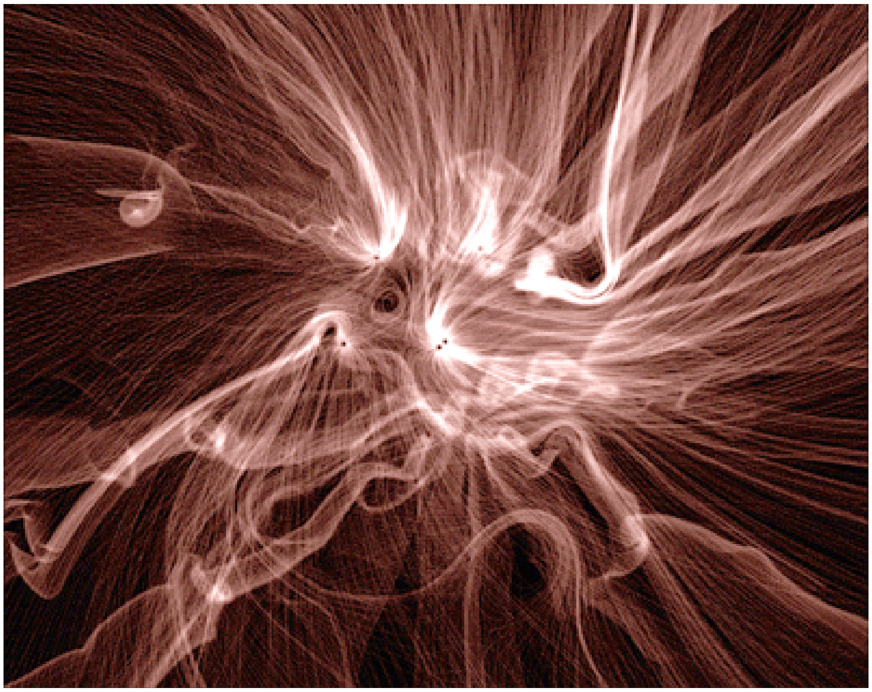


Fig. 9. Golan Levin, Still Image from “Floo,” 2000 [6]

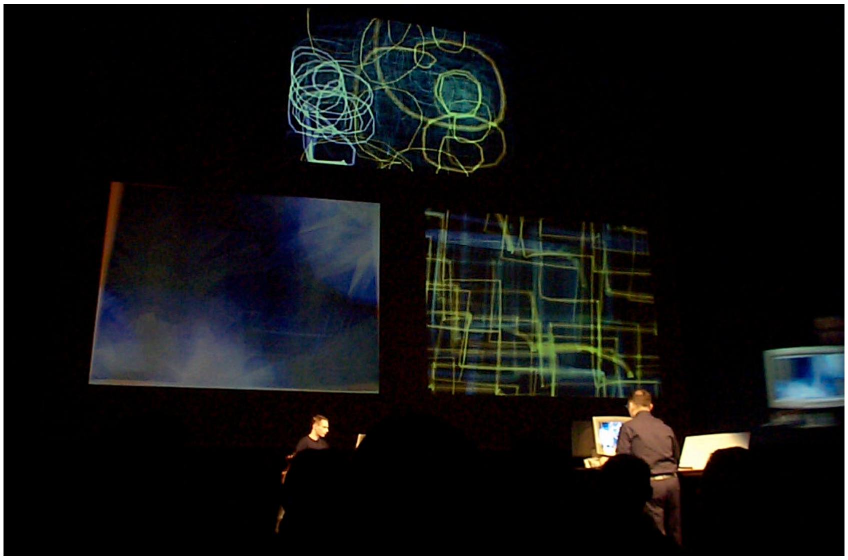


Fig. 10. Golan Levin, Still image from “Scribble,” 2000 [6]

All five of these systems were performed simultaneously as a whole in a performance named

Scribble (see Figure 10) at Ars Electronica in 2000 [6].

Artist Lynn Pocock-Williams has also explored the integration of sound and images through computers. Pocock-Williams feels that “visual music involves a very complex union of images and sound in order to create an experience that is somehow more expressive than the sum of its parts” [11]. Her desire to use the computer as a customized tool to generate and analyze art became the impetus behind her software creation. This software analyzes various aspects of a musical score and generates a corresponding expressive animation. Pocock-Williams’ research yielded two projects. The first was a set of three “predetermined” animations, created using traditional computer graphic techniques. She would repeatedly listen to the music to understand the mood and overall quality of the piece, and then create animations reflecting the intensity and tempo of the music. Since these animations were pre-edited and not generated in real time from the music itself, they served as an experiment and learning process for the implementation of the “automatically generated animations.” The second project in Lynn’s research was the creation of a computer system analyzing the pitch and duration of sound to automatically generate animations. This process consisted of five major steps. The first step required that the music be entered into the computer and then extracted for analysis. The second step was the analysis of the music in 10-second segments, which was converted into “simplified pitch-time graphs”. These graphs were reviewed in step three and a corresponding animation was assigned, chosen at random from a pre-categorized group of animations. A library of 10 second animations, “classified by predominant pattern and overall visual qualities, is initially built and then [selected from] during the generation of an animation” [11]. This process of analyzing the music segments and choosing an animation continued until the music is finished. The order of all animation segments was stored in a final animation file and then printed out to tape. It is important to note that even though the animation is generated from the real-time music analysis, it is not viewed in its entirety until all analysis is complete. Therefore, an animation is generated that will correspond to the music but it must be played back with the music after the analysis is complete.

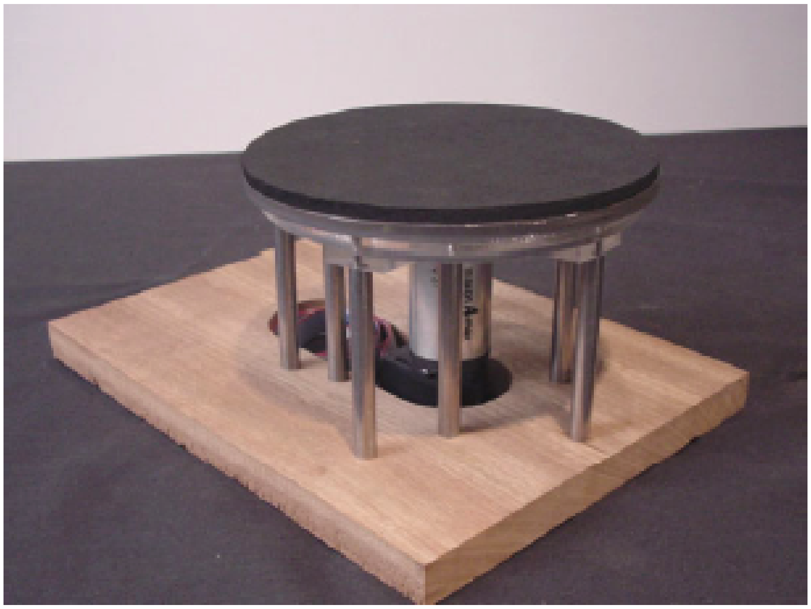


Fig. 11. Michael Lew, “Live Cinema” [7]

These techniques for synchronizing moving images to sound in real time can be considered a modern form of interactive video editing. Michael Lew designed an “expressive tangible interface for cinema editing as a live performance” with his instrument Live Cinema (see Figure 11). It combines the ability of live video editing from a bank of images with a musical instrument, and allows for improvisational control of the performance. Lew states that “Live Cinema can be described as any performance involving the *presence of* *a human performer* manipulating moving images projected for the audience" [7]. Two concepts were created to implement the final Live Cinema performance tool. The first was Image Canvas, a large display area that allowed the user to touch and move around images to create a live, gestural video editing experience. The second, Video Drum, was a circular disk-based time controller that incorporates speed control through touch, frame-by-frame position control, and hit sensitivity to control physical playback of a selected video [7].

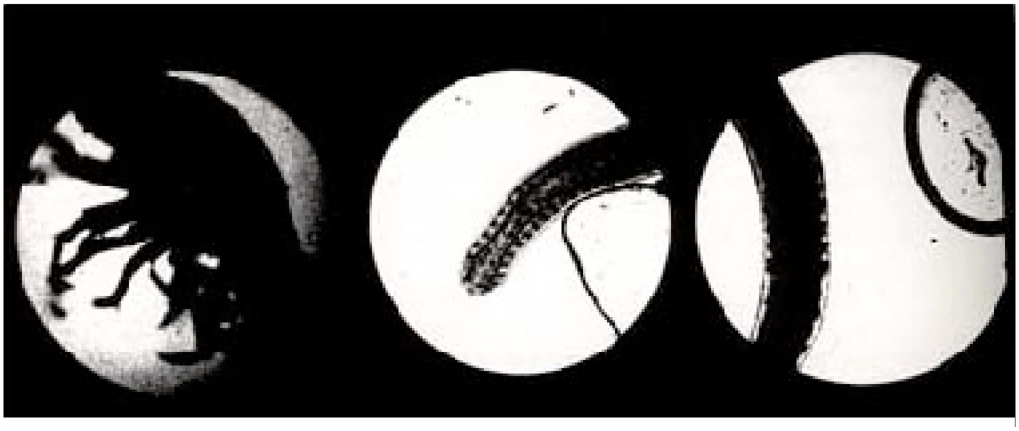


Fig. 12. Mark Boyle, Still image of insect projections, 1960's [3]

This type of real-time editing can also be seen at clubs and concerts where VJs use live video editing software to manually mix visuals for live music performances just as disc jockeys (DJs) mix music. During the 60's, psychedelic artist Mark Boyle used a standard overhead projector to mesmerize crowds with magnified images of chemical reactions, squirming insects and shapes made by body fluids [3] (see Figure 12). Others created light shows with prisms, or projected lasers onto the surrounding landscape. Eventually the affordability and effectiveness of video projectors and multimedia computers allowed VJs to produce and edit more elaborate visuals in real time. Coldcut, a VJ company, released a program that "integrates video and sound, allowing Coldcut's video clips to be triggered and scratched" [3] in real-time. An example of this is their song "Timber" where they use video clips of axes, chainsaws and bulldozers as subject matter. Other artists create their own editing programs with software such as Max/MSP/Jitter, Resolume, Arkaos and Onadime, as well as open source tools like PD, VVVV, and Apple's Quartz Comoposer. VJ performances serve as backdrops for venues from rock band concerts, nightclubs, to classical music concerts. In 2002, several artists from various professions collaborated to produce *Sun Rings* (see Figure 13), a multimedia production featuring the Kronos Quartet, a choir, and both audible wave frequencies and images from space. The wave frequencies were generated by various atmospheric events occurring in space producing eerie whistles and tones. Artist

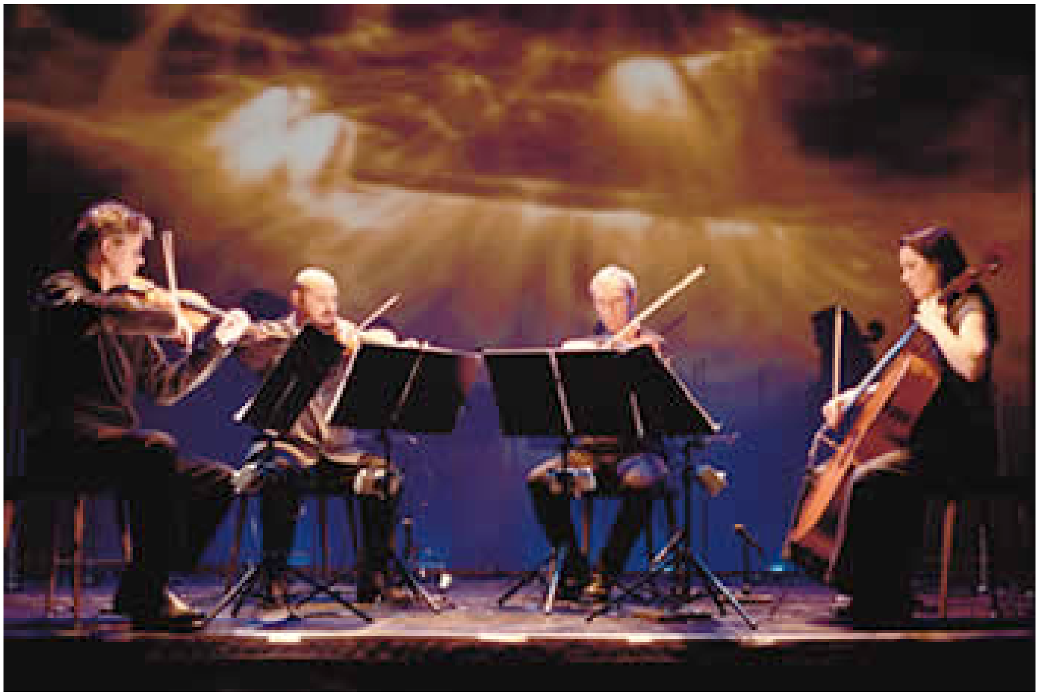


Fig. 13. Kronos Quartet and Willie Williams, "Sun Rings" 2002 [19]

Willie Williams, whose visual collection was provided by NASA, used moving images that were raw and “extremely rough in quality, but their authenticity [conveyed] enormous emotional power” [19]. Williams and the other artists choreographed the imagery to each segment of the 10-movement performance, allowing the complexity of images and visual motion to increase with the music. It was important to present these powerful images so they “blended into a show that was, after all, supposed to be about the music” [16].

CHAPTER III

WORKING PROCESS

One of the most important components in generating a successful audio-interactive visual performance is the content of the imagery used. The portrayal of the desired tone, rhythm, and mood of the piece through appropriate subject matter is crucial to enhancing the music. When deciding what type of visual imagery to accompany a selected song, artist Lynn Pocock-Williams would "listen to the music many times until [she] had a sense of the overall quality of the corresponding images" [11]. This ”listen and response” technique is a key factor in deciding how to visualize the music, and played a major role in the implementation of this project. The chosen music, *Short Ride in a Fast Machine* by John Adams, is approximately five minutes long and has a consistently fast-paced tempo that builds to a climax at a few points in the music. This piece can be broken into three main sections, each building to a high point in momentum and rhythm. It is important for the artist to produce imagery that corresponds to this musical structure. Recording the appropriate images, post-processing that footage, and executing appropriate image presentation can accomplish this technique. This project uses time-based images with varying degrees of energetic, fast-paced and colorful video. The goal of this section is to build a video library that encompasses these elements.

III.1. Capturing Appropriate Imagery

The "listen and response" technique was implemented to determine what visual aspects the imagery needed to reflect various qualities of the chosen music. Analyzing the music by repeatedly listening to its musical structure, tempo and tonal accents revealed a sense of energy and quickness throughout the piece, even in the calmer moments. Because the music built to a climax several times, the imagery needed to possess a wide range of visual qualities to correspond to the changing music. One feature that the imagery needed to display was a sense of energetic movement with a broad range of motion and intensity of movement. Colorful and vibrant images containing various degrees of saturation were also important to appropriately represent the scale of musical pacing. The prominent focus of the imagery was the color and movement, and not its subject matter. After assessing these requirements, it was evident that recording amusement park rides was an appropriate solution. The ride's extreme range of motion and energy coupled with vibrant and flashy color would easily reflect the music.

After deciding on the subject matter for the visuals and understanding the variety of imagery needed, the next step was to shoot the video. An important factor to consider was the video resolution to be used. Although recording at the highest resolution possible is always desired, this footage was recorded at 640x480 resolution because of its use in the interactive program. This resolution size allowed the interactive program to function properly without slowing down the computer's processing speed.

Numerous video clips featuring an assortment of rides were captured to ensure a variety of motion, color, texture and energy when editing the final performance. These clips implemented various framing techniques. Some shots were close-up details of the ride (see Figure 14), which abstracted the content of the image and created a non-representational video. This detailed framing allowed the motion of the video content to cover the full frame of the shot, passing beyond the boundaries of the frame. This style of framing created a very colorful and motion intensive video clip that was used to represent the bass instruments with a deeper and stronger tone. Some clips were medium shots that focused on a broader framing of the ride and its motion (see Figure 15). With this type of shot, parts of the ride moved in and out of the frame and concealed any motion beyond these framing boundaries. This method also generated clips with extreme motion because a broader range of the motion was captured. Depending on the amount of color and captured motion of each ride, these clips represented a variety of low-to-mid range instruments and helped build the overall tone of the performance. Other clips employed wide-shot techniques (see Figure 16), capturing the entire ride and its full range of motion. The rides in these clips did not break the boundaries of the frame. The motion captured could also be intense and fast or smaller and more contained. With such a wide variety of color and motion, these clips were implemented to

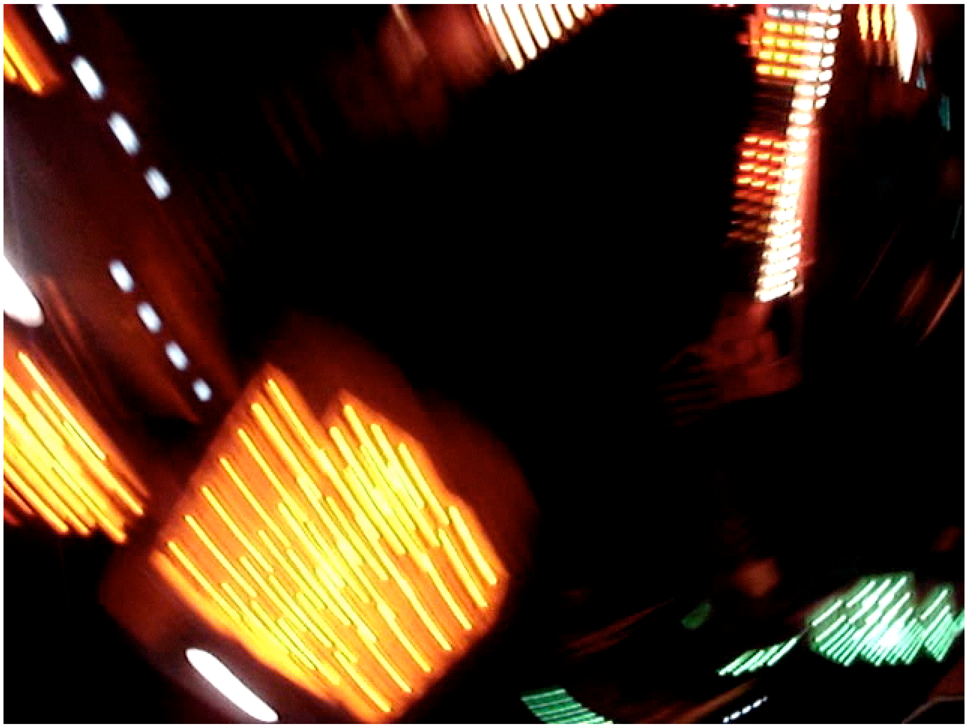


Fig. 14. Still image of close-up framing method, 2005

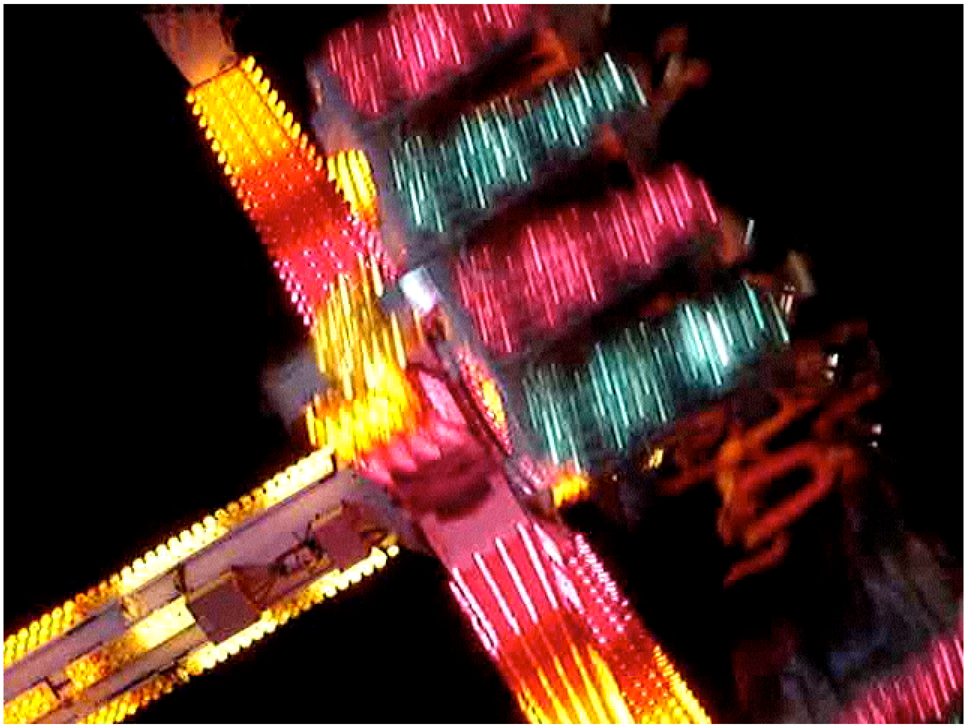


Fig. 15. Still image of medium framing method, 2005

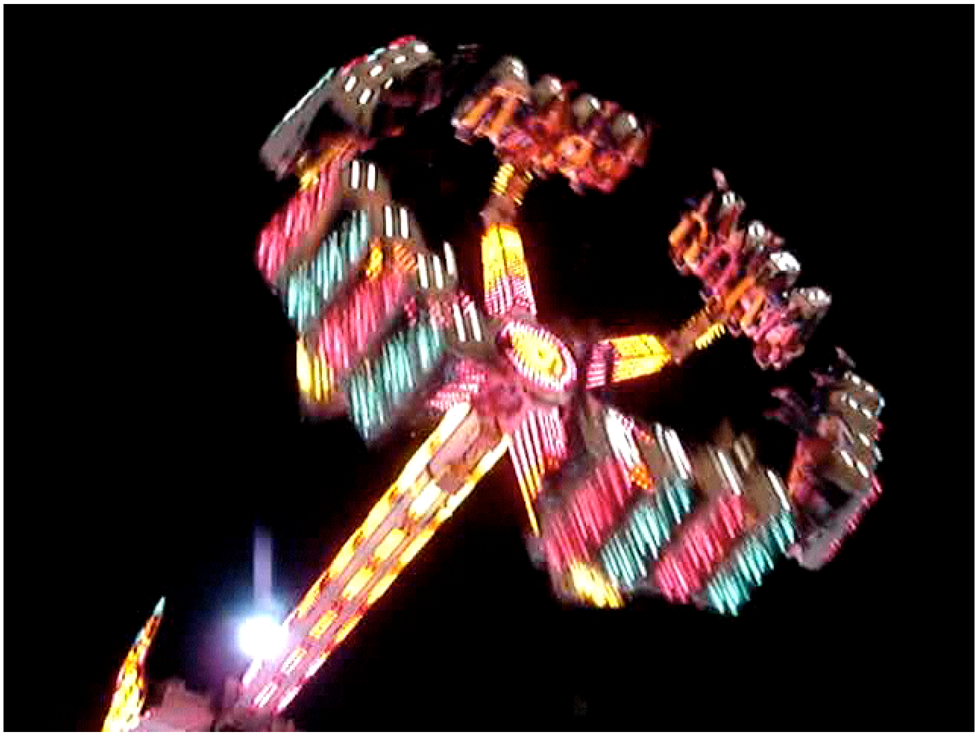


Fig. 16. Still image of wide-shot framing method, 2005

represent the entire spectrum of instruments and tones within the performance. Different styles of camera movement created a variety of motion intensities for all framing methods. The camera was either animated to follow the ride's movement, or kept stationary as the action passed through the frame. These techniques captured a broad range of motion from just one ride.

Two recording sessions were needed to obtain all the footage. The first collection of video images, consisting of large swinging mechanical arms, spinning machines, roller coasters and other such rides, was recorded at the State Fair of Texas in October of 2005. A second collection of video images was acquired from a traveling fair in College Station during that same month. This completed a broad library of video footage, which was then post-processed to create an even greater amount of clips to use.

III.2. Post-Processing the Video Footage

Once collected, the video footage was heavily processed, abstracted and manipulated such that the imagery presented itself as neither a visually representational nor a narrative piece. This was necessary because it helped to build a collection of imagery that had the energy and pace that was desired for the presentation but was not literal to the original images. Through the abstraction of the images, the audience members could make their own interpretations of the visuals and let the performance be a purely entertaining experience. To achieve this abstract video, the software Adobe After Effects was implemented. This software allows for the manipulation of video through a variety of filters, image-layering tools, warping tools and many other 2D effects. Emphasis on variations of color saturation, changes in image texture, image layering to create kaleidoscope-like effects, and changes in intensities of motion throughout the performance developed the desired energy. The method of abstracting the video images was a working process that was constantly evolving. New techniques were formed with each new clip created.

The first method used to abstract the video images was implementing a layering technique that created a distinct and complex image from the original. This was a straightforward approach, keeping the footage true to color and speed while layering multiple video clips. Importing several clips into the composition, rotating their orientation, and using blending modes to create a seamless image generated a unique overall motion (see Figures 17 and 18). The individual video clips became a unified entity that seemed to fold and undulate within itself, creating a kaleidoscopic effect. This technique was repeated using different video clips, and each composition yielded very different abstractions. Adjusting the blending modes for each layer caused interesting interactions between the layers. Each blending mode created a different effect depending on the image it was affecting and how much hue, saturation and contrast the image contained. Changing the scale of each layer also added a unique dimension to the composition. This layering technique became the basis for all the abstracted clips, and was the most important factor in creating a unique and non-representational image.

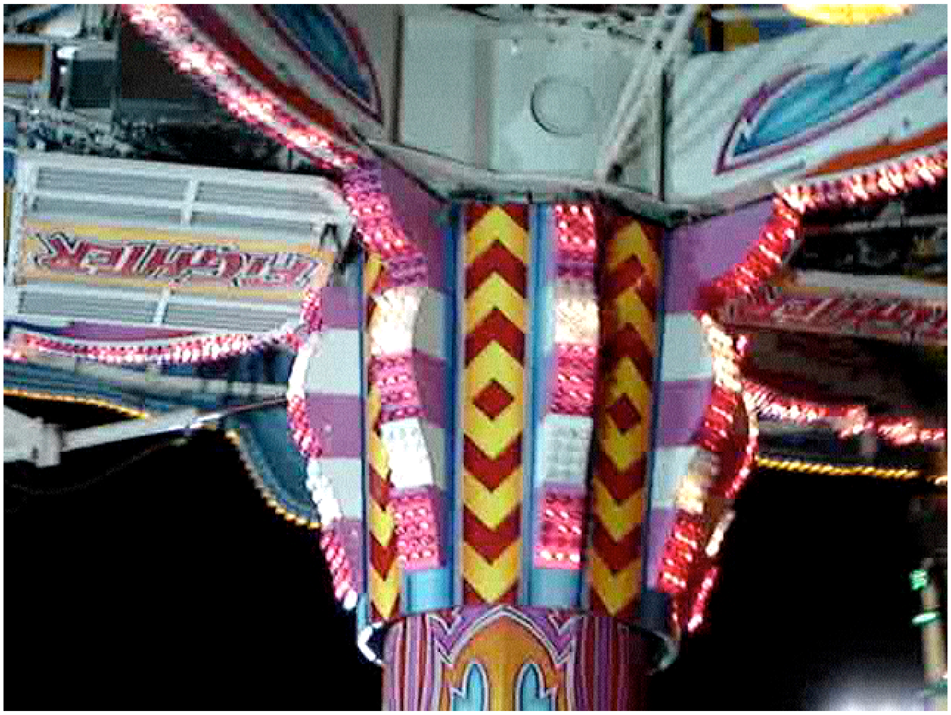


Fig. 17. Layering technique – Original still image, 2005

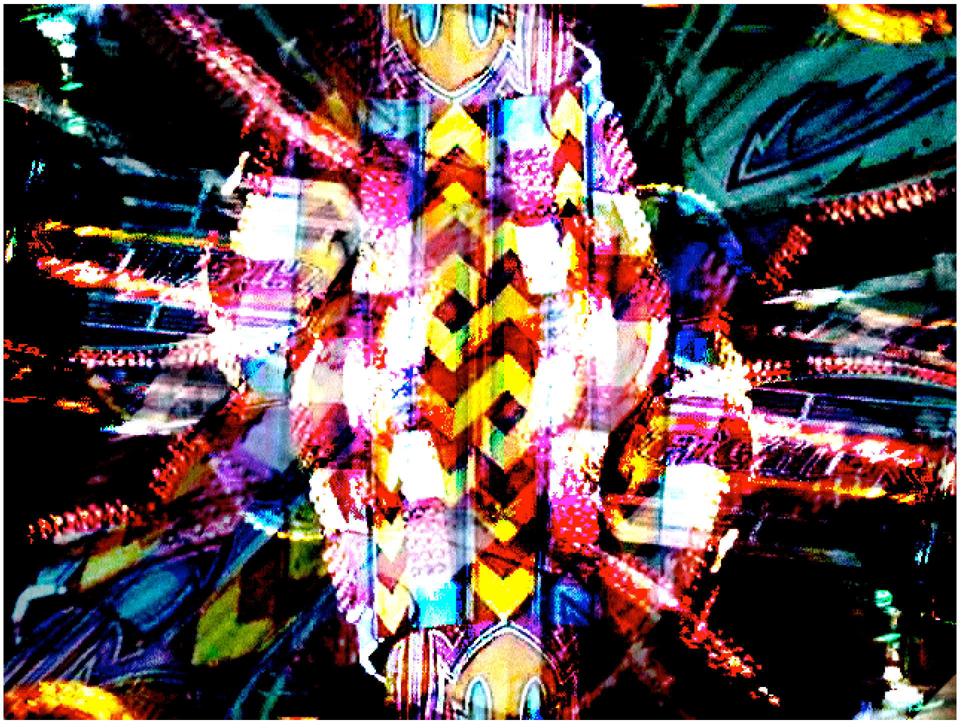


Fig. 18. Layering technique – Final still image, 2005

Another approach consisted of applying video filters that adjust the color and texture of each image layer. By changing the color saturation, hue and texture, more depth was created that gave a stylized look to the piece (see Figures 19-22). Using the Adjust filters such as Hue/Saturation, Brightness/Contrast, and Channel Mixer pushed the color combinations and intensity to an extreme. The Stylize filters such as Glow, Brush Stroke, and Color Emboss added texture to the image, causing the original source footage to become unrecognizable. Adjusting any of the filters within After Effects had a great impact on the work by causing the image to become vibrant and intense in hue and texture. The combination of filters and color adjusting techniques were endless, and allowed for efficient creation of stylized and abstracted imagery.

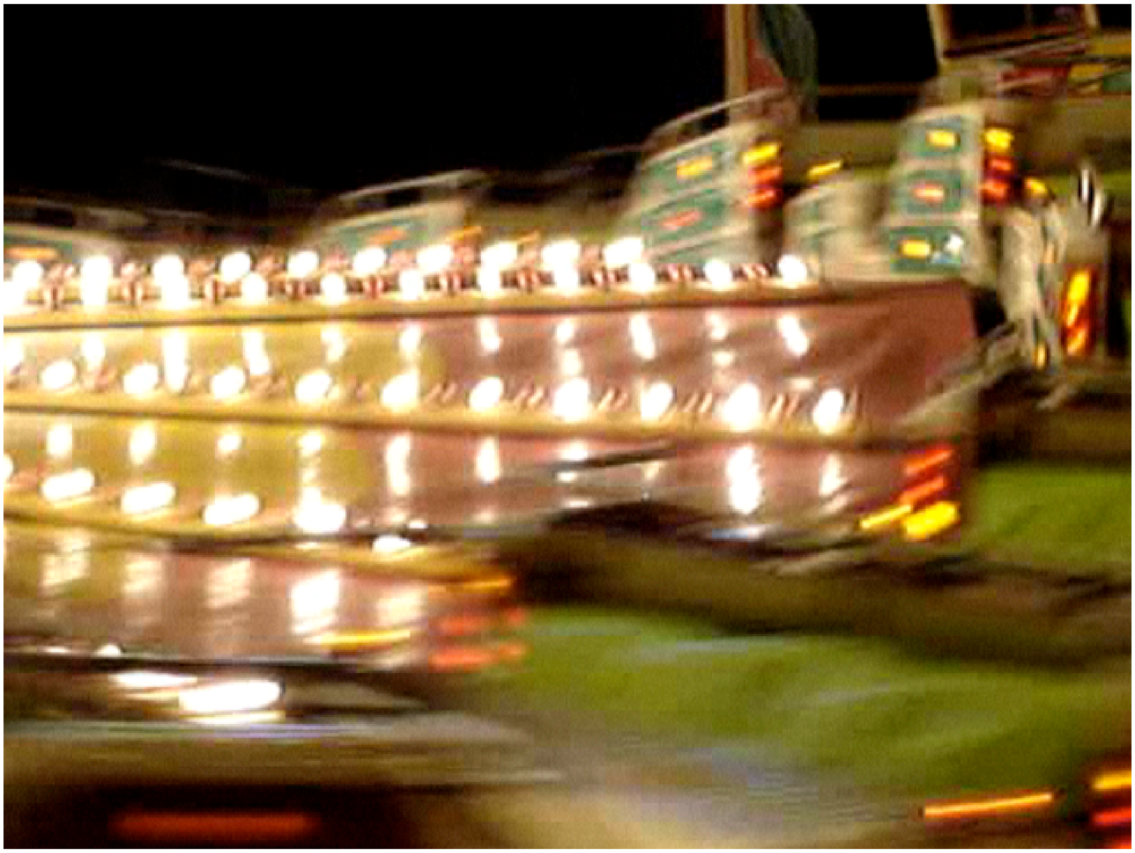


Fig. 19. Filtering technique – Original still image, 2005

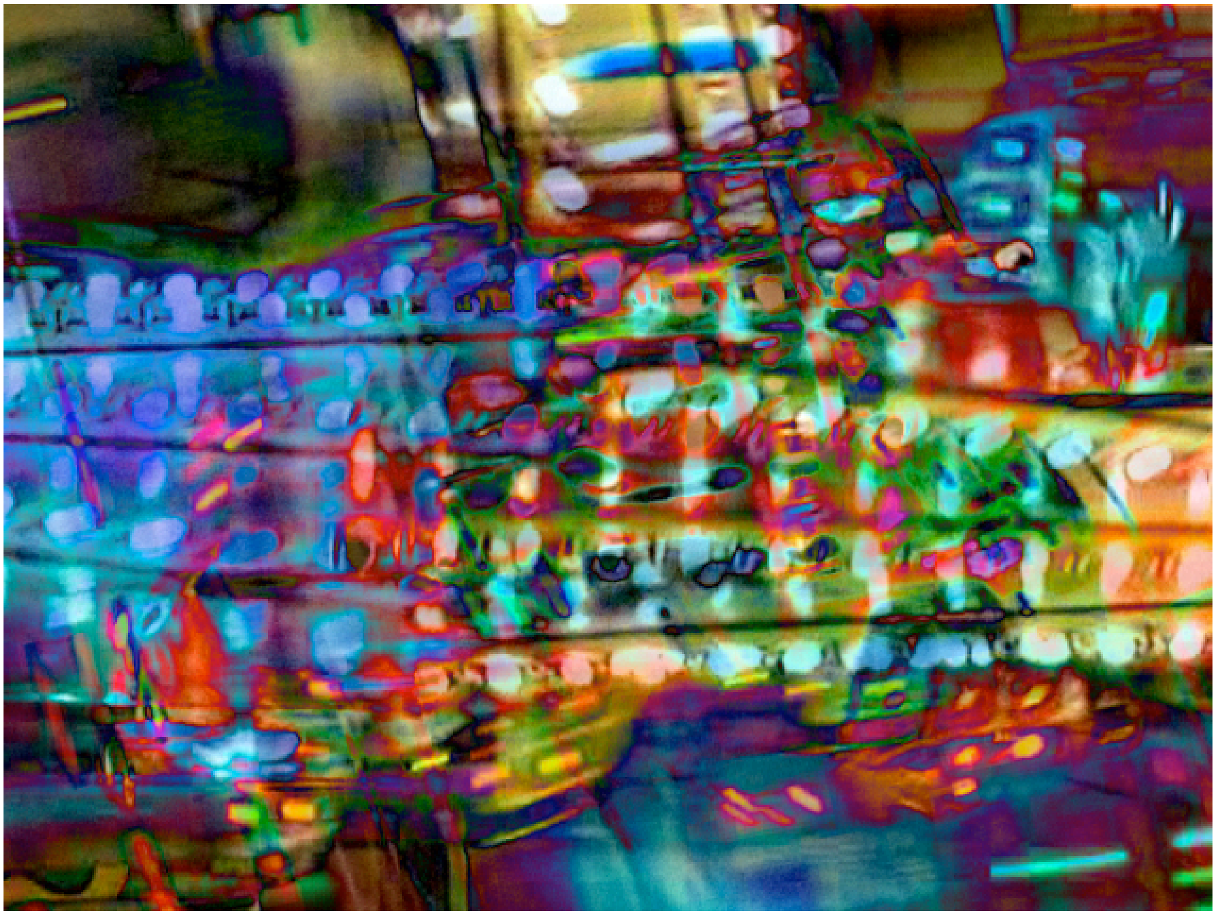


Fig. 20. Filtering technique – Final still image, 2005

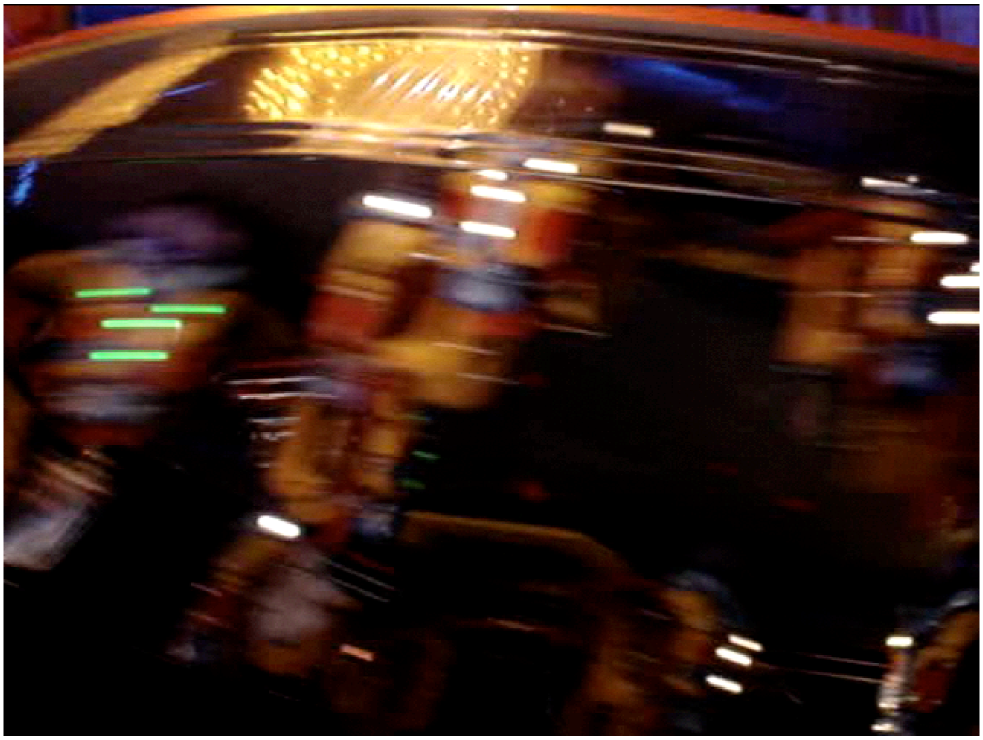


Fig. 21. Filtering technique – Original still image, 2005

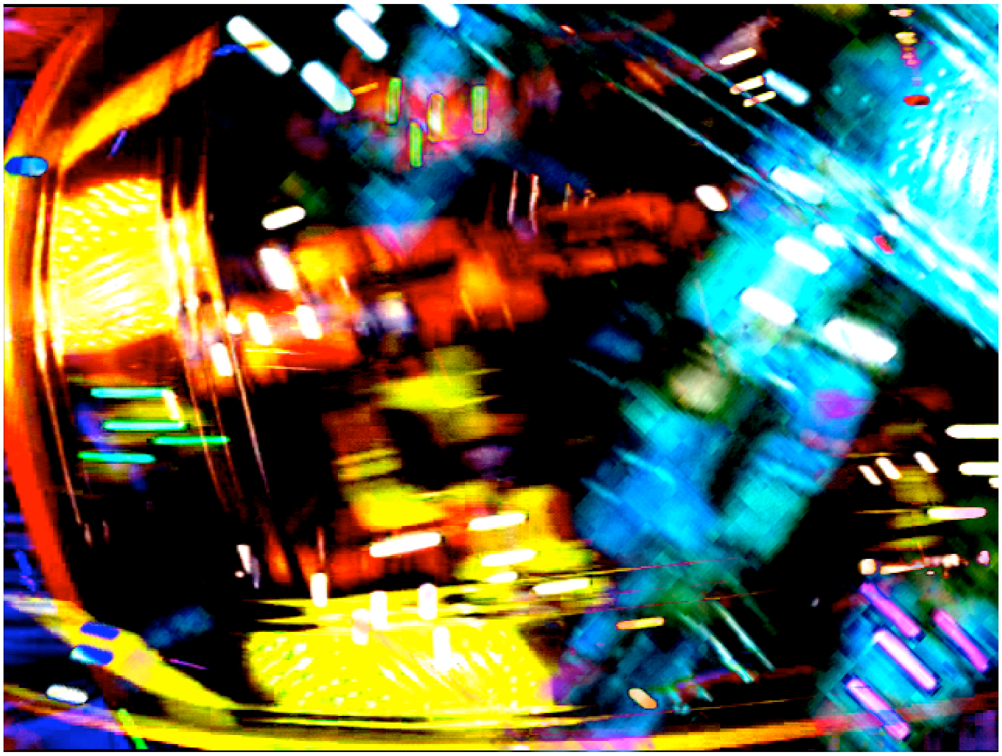


Fig. 22. Filtering technique – Final still image, 2005

Warping each layer was another procedure that added variety to the compositions. Using the Distort filters such as Liquefy, Smear and Twirl caused the video motion to became wavy and dreamlike (see Figures 23 and 24). Source footage was manipulated with the distort tools by moving the pixels on the screen until the image became deformed. As the video played, the movement of the rides became distorted as it followed the image deformations.

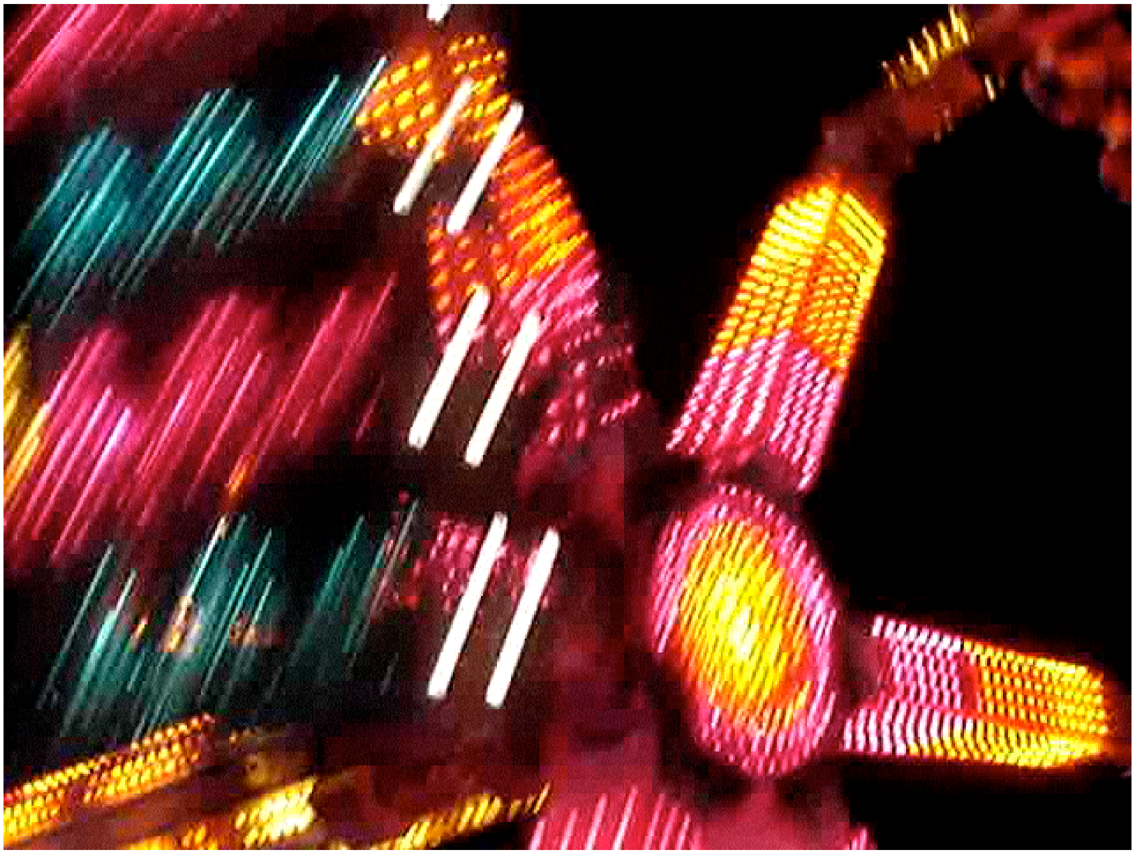


Fig. 23. Warp technique – Original still image, 2005

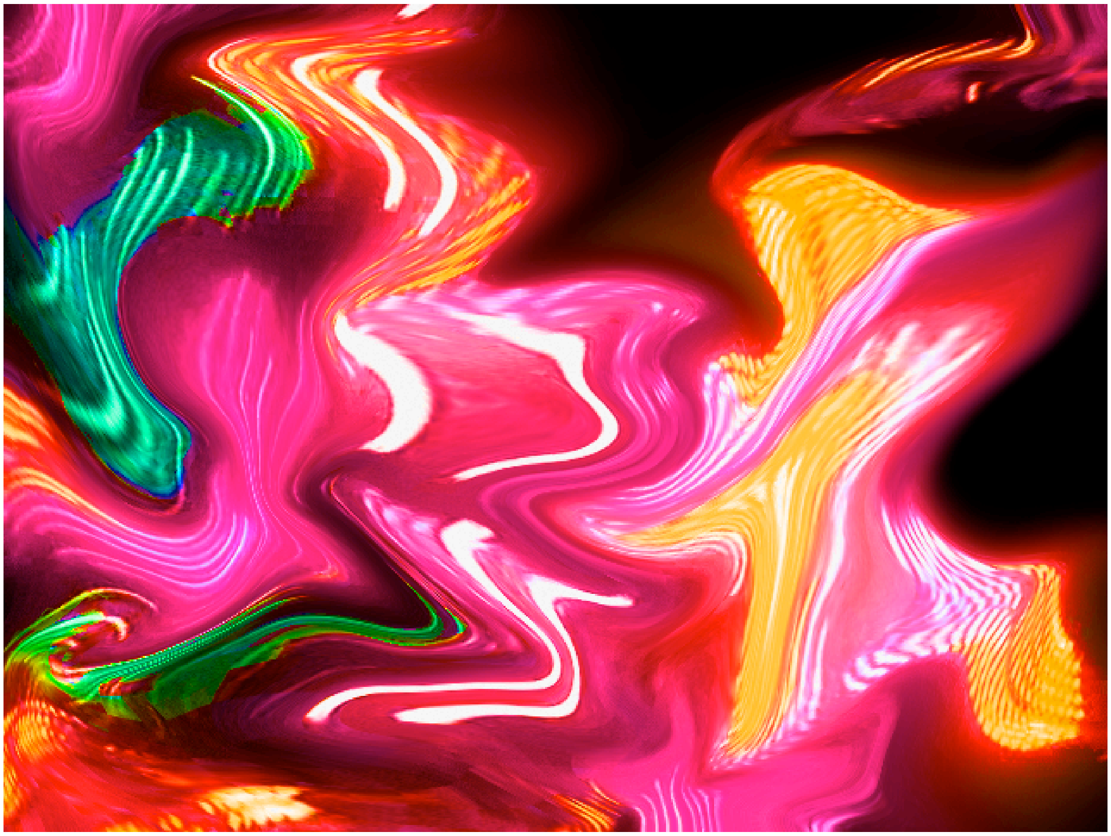


Fig. 24. Warp technique – Final still image, 2005

Changing the playback speed and animating the orientation of the layer was another approach that caused a distinctive, disorienting effect. When all of the layers were playing at the same speed, the abstracted clip was mesmerizing yet consistent in tempo. Animating the orientation of each layer and varying the playback speed added a level of complexity and unpredictability. The first attempt at this concept was animating each layer to rotate at a different speed. Depending on the amount of variation between layers, the abstracted clip would begin with a unified look, but quickly become unsynchronized as the difference between each layer's rotation speed became apparent. The next attempt was to adjust the playback speed of each layer so that the overall motion was initially varied. This technique proved useful in creating intricate motion when it was used in conjunction with the layering technique because of the interaction between each layer. For example, if there were four layers in a composition, one of the layer's speed might stay at 100% while the second layer's speed was adjusted to 75%, the third layer's speed was adjusted to 40% and the fourth layer's speed became 30%. When these videos were played together, the variation in the playback speed created an illusion of more complicated movement.

Most of the abstracted compositions consisted of all the same video footage, yet combining different source footage was also successful. Using different source footage created multifaceted movement unlike the other techniques, and was one of the fastest methods to generate interesting results. The key factor in creating the abstracted video image was applying blending modes to each layer. Once an effective blending mode was chosen, the separate layers were revealed as one unique image, and the complexity of the new motion was apparent. The difficulty was finding the right combination of footage that not only compliments but also contrasts each other. This technique could be applied in combination with any of the above-mentioned techniques to further abstract the image.

Choosing a suitable combination of images, filters and adjustments per composition took time and many iterations of trial and error. The order in which these methods were applied produced differences in the outcome. When choosing the combination of video footage and filters it was important to remember the pace and look that was needed for that clip. If many video clips were created that were high in intensity of motion and color, the next few clips created would be lower intensity to maintain balance in the video library. Over eighty abstracted clips were produced from less than thirty original clips.

III.3. Categorizing and Presenting the Imagery

Abstracting the original footage allowed for a multitude of images to be made that could be categorized into various sections and later selected at the appropriate time. Since the music could be broken down into to three main sections, the visuals were categorized into three sections as well. These sections were High, Medium and Low, which corresponded to the intensity of motion, color and detail of each clip. High footage consisted of images that were sparse in color with a quick yet contained motion that did not move all over the screen (see Figure 25). It was important that the motion did not reach past the boundaries of the frame, producing a sense of control within the video. The clips had a delicate and sparkling quality

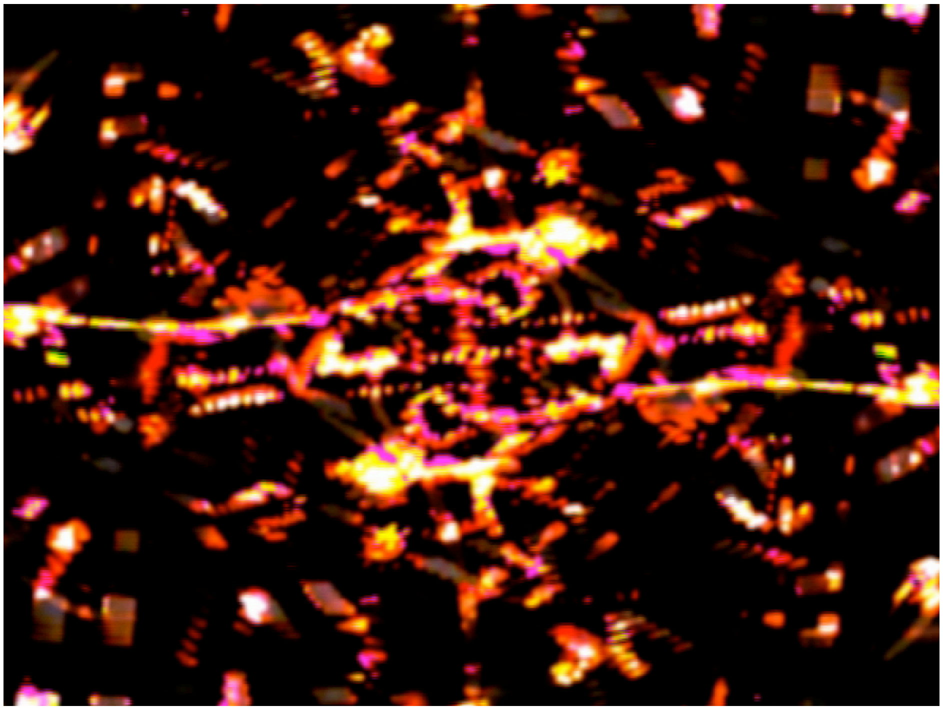


Fig. 25. Still image of High Footage, 2005

that appeared delicate and corresponded with the treble instruments sounds such as the flute and cymbals.

Medium footage covered a broad range of attributes (see Figure 26). The majority of the screen was vibrant with a wide spectrum of color and contrast. The motion was medium to high paced, and moved from one side of the screen to the other without severely passing the boundaries of the frame. This type of framing was important because it clearly showed the movement of the ride and generated a variety processed video clips with the required pacing for this section. This Medium section was the largest of all sections and its footage was used as the underlying video throughout the performance. The Medium footage was not directly correlated to a particular sound or frequency range but helped build the arc of the performance.

All clips that corresponded to the low-to-mid range drum sounds such as the tympani

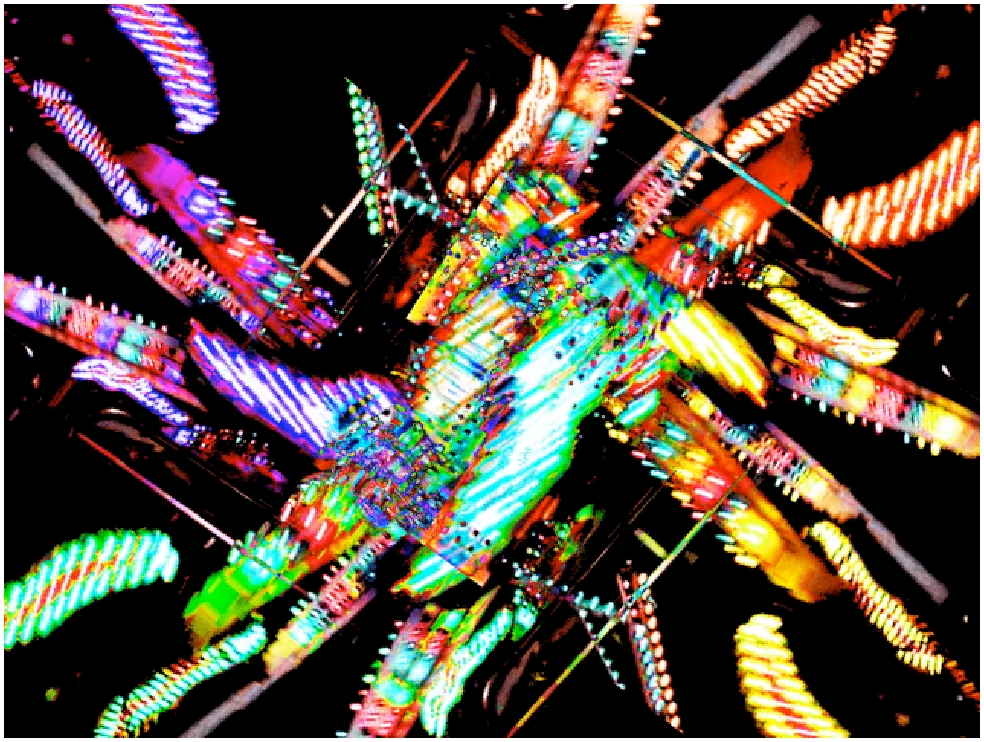


Fig. 26. Still image of Medium Footage, 2005

or snare drum were placed in the Low category. These clips were intense in color and motion (see Figure 27). The movement on the screen was very fast paced, covered the entire frame, and often reached beyond the boundaries of the screen. Allowing the motion to travel beyond the frame's edges was important in conveying a sense of extreme movement and excitement. The imagery within this section had a deep and powerful presence with the most vibrant and saturated color filling the entire screen. The combination of color, motion and conveyed energy was imperative to each section, and defined how the video within that section was implemented in the performance.

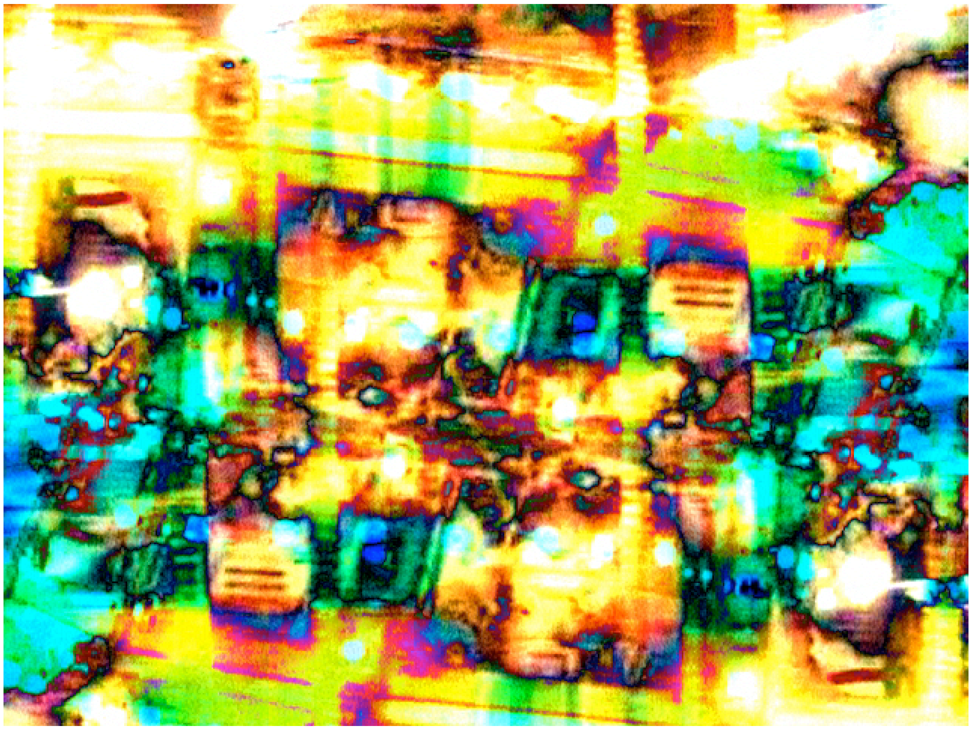


Fig. 27. Still image of Low Footage, 2005

Controlling the organization and appropriate use of the imagery assisted in creating the visual arc of the mood and rhythm for the performance. As the music built with tension and excitement, so did the visuals. The first section of the music introduces the performance pace to the audience through a simple structure. Beginning with a staccato musical rhythm, the musical intensity progressively increases by layering the instrument sounds. The video imagery corresponds to this structure by first revealing desaturated video with content that is more recognizable in comparison to the video revealed later and slightly filtered. As more instruments enter the performance, the color and vibrancy of the images increased. The background color of these images remained a solid color so that the focus was on the motion of the abstracted ride and was not overwhelming for the viewer. Real-time filtering, which distorts the final output video into various predefined shapes, was used to accentuate high and low frequency activity within this section. The section finished with a swell in volume, use of instruments, and intensity, which the imagery reflected by increasing color saturation, texture and motion. The real-time filtering became the focal point at the end of this section as it was used to accentuate the increased use of tympani drum or tenor drum. The motion within the video images for this section remained constant as the color and saturation increased in vibrance and use of variation with the music.

After the crescendo of the first section subsided, the music returned to the original, slower pacing and began to build again in intensity. Since vibrant color had been introduced near the end of the first section, the second section began almost immediately using color. Responding to the rhythmic bass drum attacks that grew in intensity, the amount of color and liveliness within the imagery increased. The musical score developed into a frenzied arrangement of discordant sounds as every instrument played at the same frantic time. To reflect this energy, the visuals became a compilation of flashing color and furious action across the screen. By the end of this section the imagery was meant to overwhelm the viewer with intense color and motion in equivalence to the music's energy and chaos.

The music in the third section contrasted the previous two sections by accentuating only the brass instruments and changing the tone and rhythm of the music, which called for a different approach to the visuals for this section. The first two sections incorporated all of the instruments into the musical structure, accentuating sections of the symphony at various moments. The rhythm and overall tone of the second section was a continuation of the first section with variations of instrument presentation and an increase in musical intensity. In contrast, the third section focused on the brass instruments with sporadic interjections from the percussion and treble instruments. The music built with a crescendo as in the first two sections, but conveyed a different mood and rhythm that indicated the end of the performance. To reflect this change, the video imagery was manipulated in a different manner from the previous footage. Using the warp techniques found in After Effects, the imagery was affected to display a dreamlike and liquid quality. The colors cascaded across the screen in a flowing motion. Delicate looking video clips from the High section appeared on top of this footage each time a high frequency instrument such as the flute or cymbals played. Like the first two sections, the music grew in intensity, as did the imagery. Vibrant radial and kaleidoscopic filters accompanied the brass instruments' rise in volume, accentuating their presence within the music (see Figures 28 and 29). Implementing these filters, which had not been used before, added to the contrast between this section and the others.

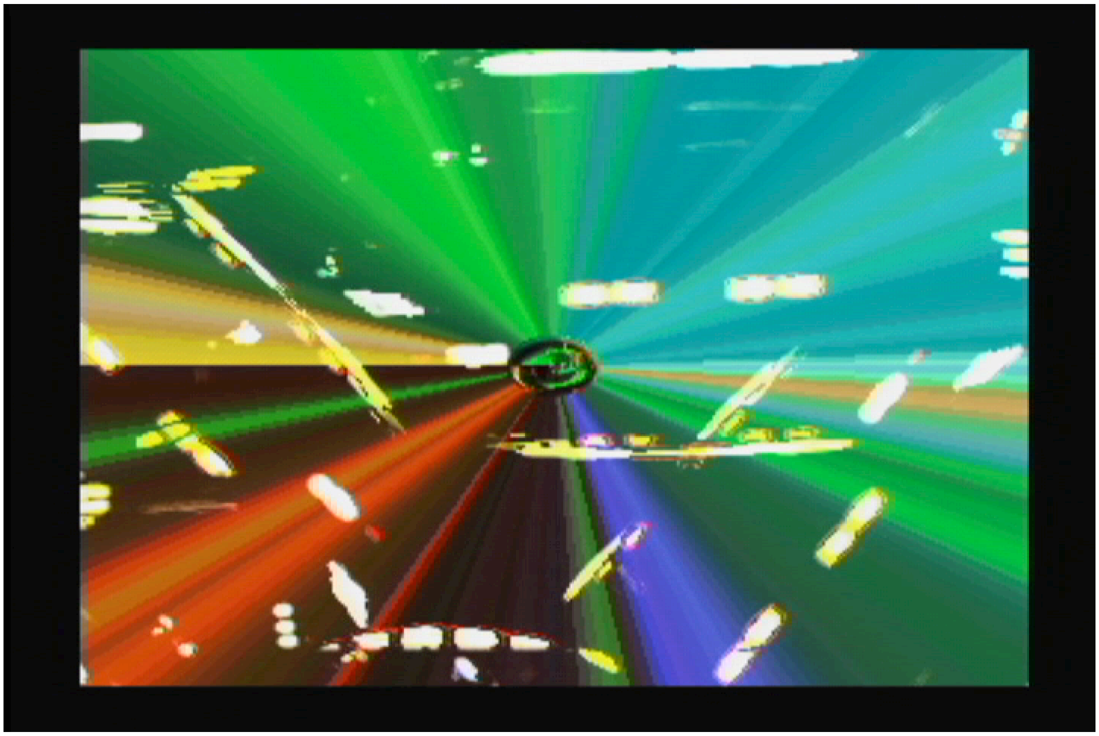


Fig. 28. Still image of Radial Filter #1, 2005

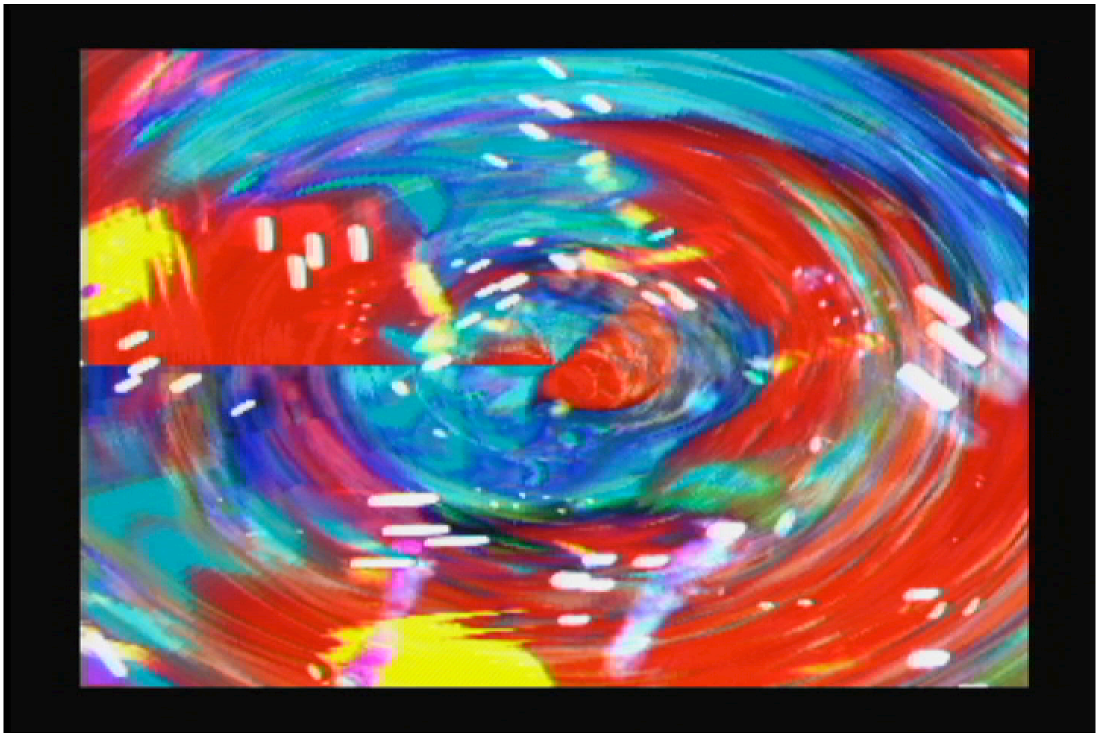


Fig. 29. Still image of Radial Filter #2, 2005

The bank of categorized video footage was integrated into the interactive program and thoroughly reviewed for suitable audio-visual correlation. As the images were created and categorized, they were viewed with the music until they complimented the tempo and mood of the piece. If the audio-visual correlation was not clear, the video was further manipulated or re-categorized until satisfactory.

CHAPTER IV

IMPLEMENTATION: A REITERATIVE PROCESS

IV.1. The Interactive Program

Other than the video content, the component most important to creating this real-time visualization is the interactive program that triggers the visuals. This performance integrates real-time responsive visuals with the ability to generate impromptu interactions with these visuals during the performance. To accomplish this, the software used to write the program must permit implementation of specific and personalized programs, as well as give flexibility and control to the user. It is imperative that the software allows for real-time signal processing and manipulation of images. An easily navigated interface is helpful in proficiently creating the program and correcting any programming errors. The Max/MSP/Jitter graphical programming environment is designed to implement all of these factors to produce a personalized, interactive and responsive program for the live musical performance. Max is a graphical programming environment that handles control-rate logic. MSP is a library of audio processing objects. When Jitter is used in conjunction with Max/MSP, the ability to manipulate and trigger video in real time becomes an option because of Jitter's matrix and video processing library. This software easily enables the production of a unique, interactive program that generates real-time, audio-responsive visuals, and was a key element to producing the visual presentation performed with the Brazos Valley Symphony Orchestra.

A program built in this environment is called a patch, a series of graphical objects, each performing a specific function connected by "patch chords" to create a flow of data and control (see Figure 30). This form of visual programming makes it easy to create customized programs

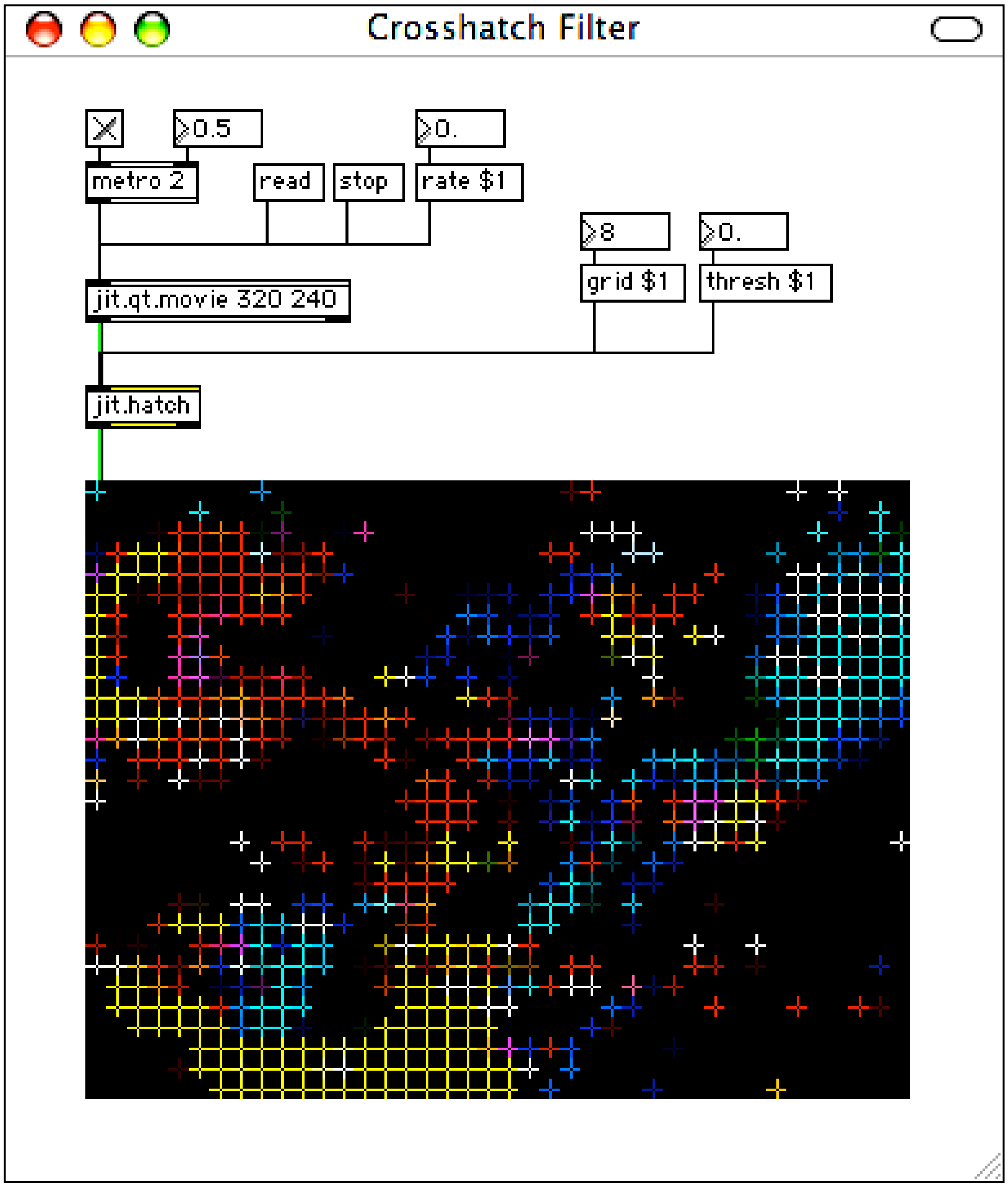


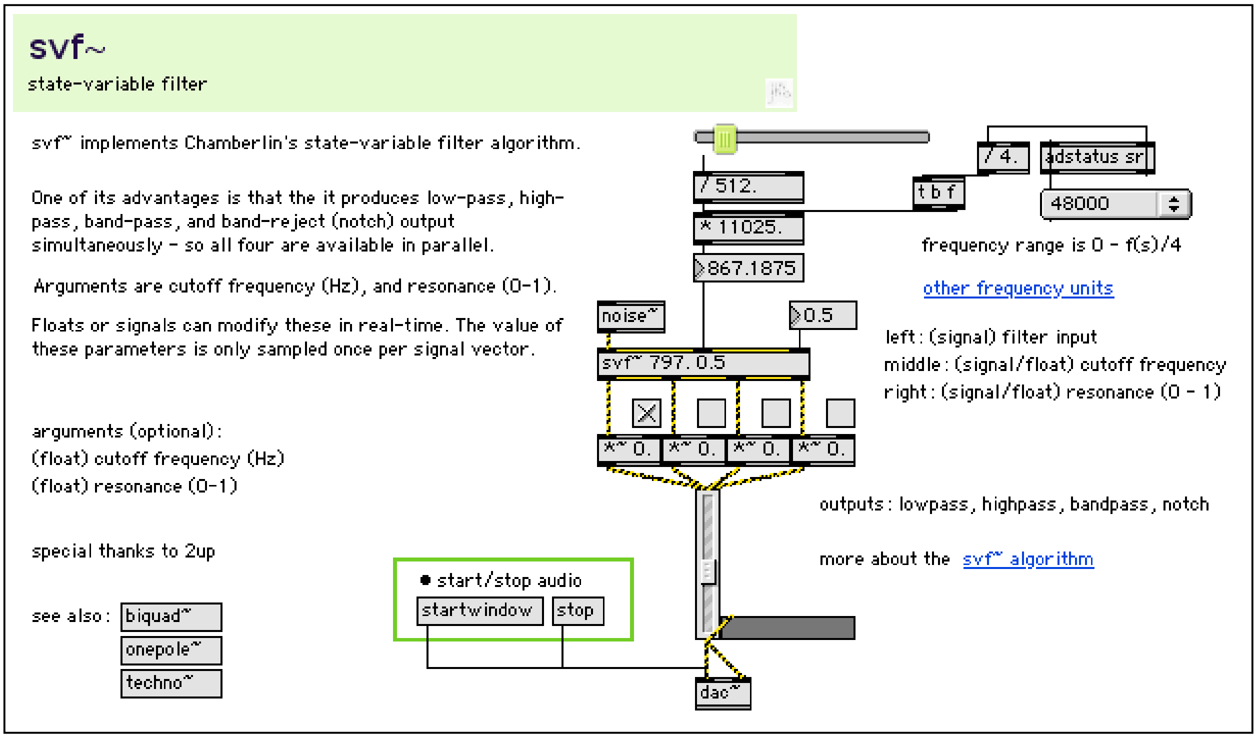
Fig. 30. Example image of a Max/MSP/Jitter patch, 2005

by eliminating the need to write hundreds of lines of code. However, the software can get very complicated and overwhelming during the development stage, and it is necessary to break down the building of the patch into simple stages.

IV.2. Audio Filtering

The first step in completing this audio-responsive program was to focus on filtering the audio, which was initially aided by simulating the live performance using pre-recorded audio input. Since this visual performance for the Brazos Valley Symphony Orchestra accentuated the music and enhanced particular sounds, it was important to filter out various frequencies and isolate certain instruments so that visuals could be triggered when the designated sound was heard. With approval from the conductor, the music was chosen to allow for a wide range of abstract visuals and arch of audio-visual intensity. A recording of the music from a commercial CD was used in this frequency filtering process to construct the system, eliminating the need for live audio during the design and testing stage. This music file is played into the program, and a frequency filtering system is created to highlight the high and low frequencies within the music.

The object svf~, found in the MSP library, enables precision filtering with filter type and parameter controls (see Figure 31). This object offers the filter types, high-pass, low-pass, band-pass and band-reject. High-pass isolates frequencies above the cut-off value, and low-pass isolates frequencies below the cut-off value. Band-pass isolates frequencies around a center frequency, and band-reject eliminates frequencies around a center frequency. Depending which filter is chosen, the undesired frequency ranges are automatically eliminated from the audio output. Further adjustments can be applied to the svf~ object to ensure a more accurate filtering system. One example is adjusting the cutoff frequency.



svf~

state-variable filter

svf~ implements Chamberlin's state-variable filter algorithm.

One of its advantages is that the it produces low-pass, high-pass, band-pass, and band-reject (notch) output simultaneously - so all four are available in parallel.

Arguments are cutoff frequency (Hz), and resonance (0-1).

Floats or signals can modify these in real-time. The value of these parameters is only sampled once per signal vector.

arguments (optional):
(float) cutoff frequency (Hz)
(float) resonance (0-1)

special thanks to 2up

frequency range is 0 - f(s)/4

left: (signal) filter input
middle: (signal/float) cutoff frequency
right: (signal/float) resonance (0 - 1)

outputs: lowpass, highpass, bandpass, notch

more about the svf~ algorithm

Fig. 31. Max/MSP library, svf~ object definition, 2005

By adjusting this value, the svf~ object is focused to a more narrow and specific frequency range, eliminating unwanted frequencies beyond that cutoff value (see Figure 32). Treble instruments, such as the flute and cymbals, are isolated with the high-pass filter by eliminating the low frequencies. Removing the high frequencies from the output with the low-pass filter emphasizes low and mid range sounds, such as tympani and snare drums. Experimentation with these filters and the designated music revealed that four svf~ objects, two high-pass and two low-pass, were needed to produce a thorough analysis of the music and range of selected instruments. Each of these objects was refined to highlight chosen

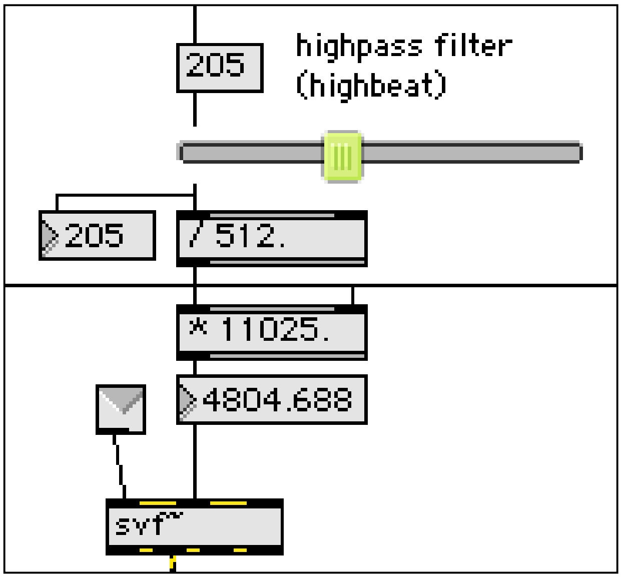
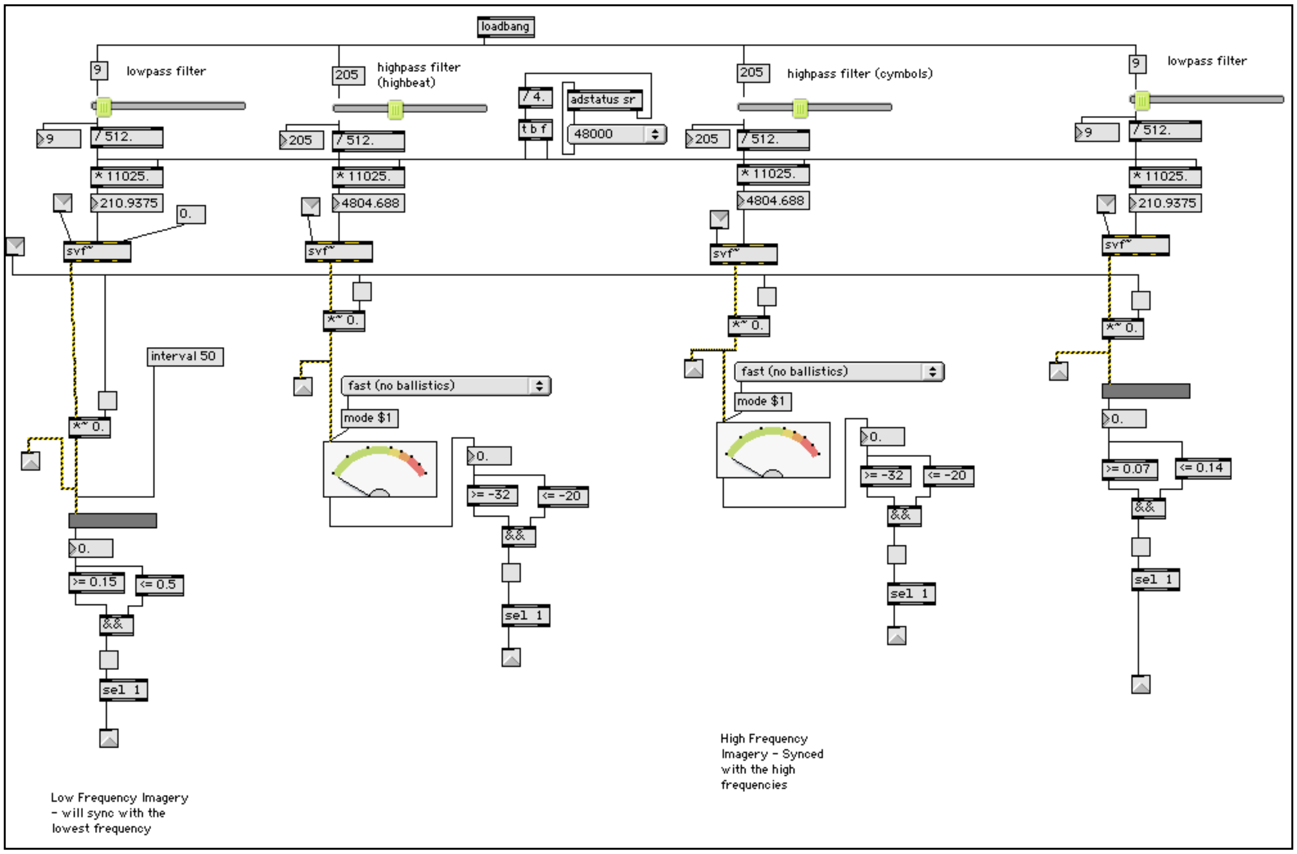


Fig. 32. Frequency cut-off slider, 2005

instruments and moments within the music.

The filtered audio is then sent to a threshold testing system, which further defines the accuracy of the audio-visual response. All parameters were defined by the artist to achieve effective audio response and filtering flexibility per instrument. The filtered audio is connected to a meter, found in the MSP bank of objects, which measures the volume level and generates numeric information about the audio. If the level meter's output exceeds a user-defined threshold, then an event is triggered. This threshold testing ensures the software is only responsive when the instrument being monitored is at its peak volume and prevalent in the performance (see Figures 33 and 34). Setting the appropriate instrument threshold allows for accurate image response.



High Frequency Imagery - Synced with the high frequencies

Low Frequency Imagery - will sync with the lowest frequency

Fig. 33. Frequency filtering system, 2005

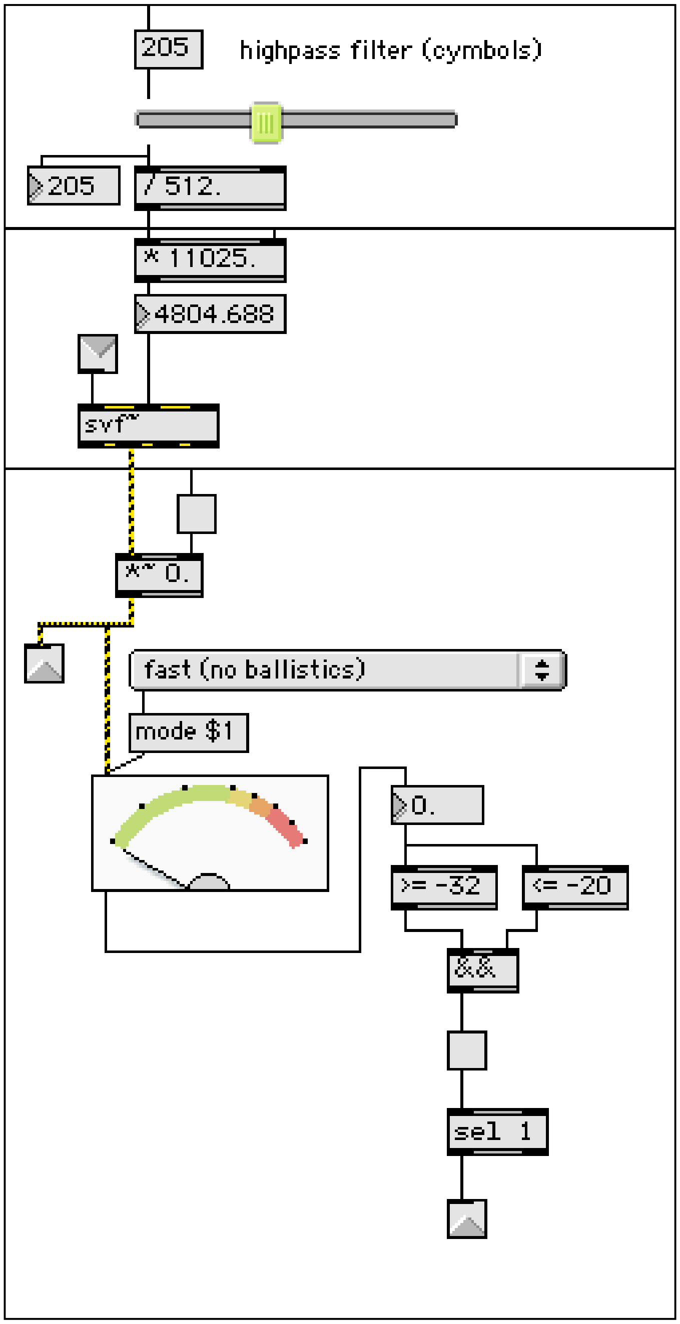


Fig. 34. Example of level meter, 2005

IV.3. Integrating the Video and Refining the Patch

Once the filtering system is responding as desired, the next step is to start integrating pre-processed video footage into the patch, which is an undulating and iterative process. Each event triggered by the audio produces a visual response. There are many factors that can create a lag in this response causing the images to un-synchronize with the music. The main concern when creating a real-time interactive program is to optimize the use of the computer processor. If the processor cannot compute all program functions and manage the video files in real time, then the audio-visual correlation is lost.

IV.3.1. Appropriate Resolution

Testing different video resolutions and file sizes with the program will help determine the most efficient way to present the processed video footage. Jitter uses matrices to process video file. Within each matrix is the ability to define variables such as plane count, and file size. The plane count defines how many color channels are included the video image out of a total of four: red, green, blue, and alpha. The user can also define the file size or resolution. A resolution of 320x240 is faster for a computer to process than a 640x480 resolution and is the optimal solution for this project. Even though it is necessary to use a 320x240 matrix within the program, the output image needs to be at least 640x480 resolution, because this is the minimum resolution requirement for all video output sources.

Two tests were created to study the effects of using different video input resolution. The first test used a 320x240 resolution video input to a 320x240 matrix. After processing, the video was output to the screen at 640x480 resolution. This produced a lower quality image than desired, because the initial video did not contain a high enough resolution for the desired output. The second test used 640x480 resolution video input to a 320x240 matrix, and output a 640x480 resolution video. This produced acceptable video quality; therefore determining that all input video should be rendered at 640x480 resolution.

To achieve maximum video processing efficiency it is also important to choose the appropriate compression settings when rendering the video. There are many compression types from which to choose from, such as Animation, and Motion JPEG A. Many of these compression types support the option to adjust the color depth (the number of bits required to draw the video 30 times per second). Choosing either "thousands" or "millions" of colors may affect file size and efficiency, but may also reduce the quality of the video. Testing these settings is necessary to determine the effective method to streamline video efficiency. For this project, Motion JPEG A was an effective choice using the settings of high quality and the default color depth.

IV.3.2. Efficiency of Video Playback

Since video files can be quite large, it is critical to determine an efficient way to load a video file, play it, and then switch to the next file without inefficiently occupying the computer's processor. This ability to show different images each time an instrument triggers an event is crucial to building tension and image variety. Even though the 640x480 video resolution is determined acceptable for processing, video file sizes can become as large as 300MB and are unmanageable. This is a problem when needing to load video files then switch between several quickly, because the hard drive and processor cannot transfer such large files efficiently.

To eliminate this issue, creating one long edit for the entire length of the music will allow for efficient image playback and audio synchronization. Since the music is approximately five minutes long, then the video edit is approximately five minutes long and the equivalent length of the music. The ability for this video edit to remain synchronized to the live music is reviewed in another section, which addresses maximizing live performance variables. This method is applied to two main video edits, each building in color, motion, rhythm and tension in correlation with the music, yet contrasting in visual content. Each file is loaded into the computer's memory for efficient playback and both are edited based on the music. During the performance, both files are playing at the same time; only one video is displayed at any given time. One video file becomes the main edit that is constantly seen, and the other video is displayed when a particular instrument is played.

Moving video is not the only imagery used during this performance, and it is necessary to determine when and how to reveal still images. During the opening moments of the music, as each instrument is introduced, blasts from the snare drum trigger colorful imagery to appear amidst the main edit of black and white video images. Since it is the beginning of the performance, the decision to reveal still and colorful images contrasting the moving low-chroma imagery is important to allow the visual presentation to build. Even though the computer can handle retrieving and displaying these images in real time because of the small file size, usually under one mega-byte, it is inefficient for this project because of the multiple processes occurring simultaneously. It is still necessary to display a still image each time the snare drum hits, so the technique of turning the still frames into a continuous video and loading it into memory is implemented. This technique is similar to the use of the main video edit but is synchronized to the music in a different manner. Each different image is one frame long within a Quicktime .mov video file. When the program is opened and the file is loaded into memory, the first image in the .mov file is queued for display. After each audio trigger from the drum, the queued image is revealed. After a two hundred-millisecond delay, the images fades away and the video progresses to the next frame. The next drumbeat triggers the new image and the process continues until stopped. This method ensures efficiency without compromising visual fabrication.

Another procedure that will slow down the computer's processing speed is showing multiple video clips on the screen at one time. During the final section of the performance, video images from the High section are composited onto the main video edit to accentuate the flute and cymbal instruments. It is important to determine what method of compositing an image onto another will be used because of the amount of time it takes to process the images, combine them, and output them to a screen. There are various methods of compositing such as luminance keying and color keying. This project uses chromakeying, which eliminates a desired color from the video image to reveal the footage underneath, giving the illusion that they are one video image. This technique allows the images to vary in blending amounts to create unique effects of foreground and background image combination.

IV.4. Efficiently Implementing Real-Time Filtering

To add to the visual breadth of the piece and accentuate high and low pitched moments within the music, filters are used that distort the image in real time. For instance, each time the flute plays, the image is fragmented into many small squares to distort the its appearance. This obvious method of image distortion helps create a direct correlation of the image to its associated sound. When choosing appropriate filters to incorporate, viewing the help patch

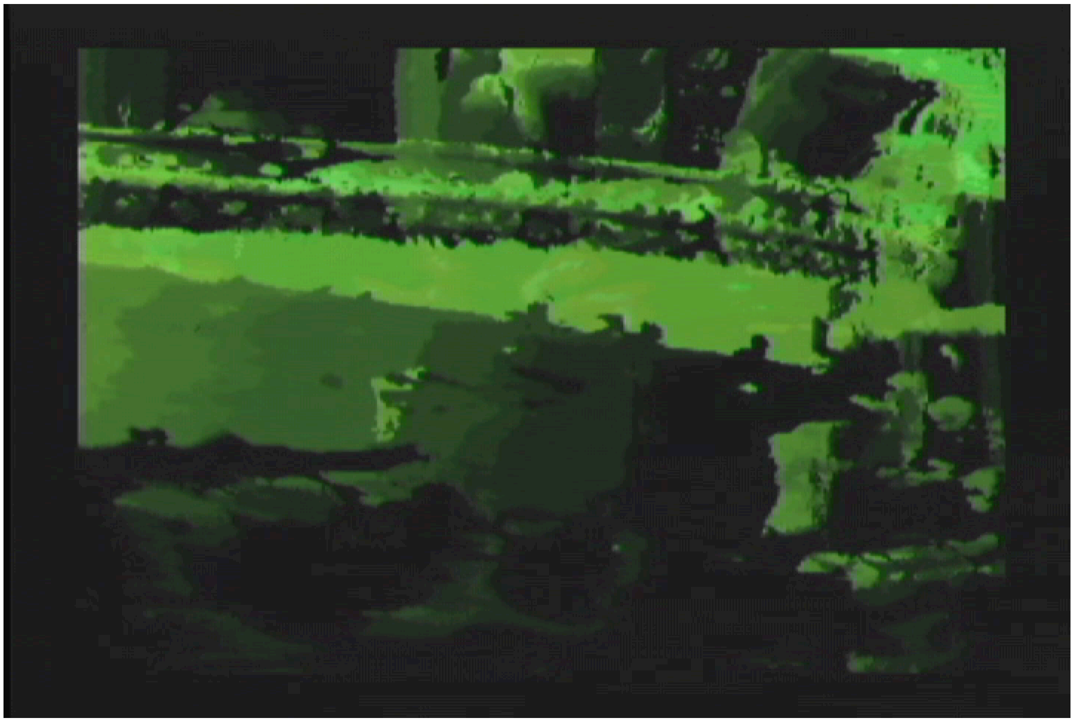


Fig. 35. Still image of jit.slide filter, 2005

of each filter and exploring its functionality in real time is beneficial. There are many predefined video filters within the Jitter library, and seven of them are implemented within this patch. All of these filters are primarily implemented during the first section of the performance. They aid in enticing the viewer to explore the connection of imagery and sound, and allow for a variety of image presentation modes throughout the entire performance.

The first real-time filter that is presented within the performance is jit.slide, which creates a ghosting effect to the image as it moves across the screen (see Figure 35). This is the only filter used that displays no direct correlation to a particular sound, but is used to introduce the audience to the imagery in an interesting manner. The filters used to accentuate the high

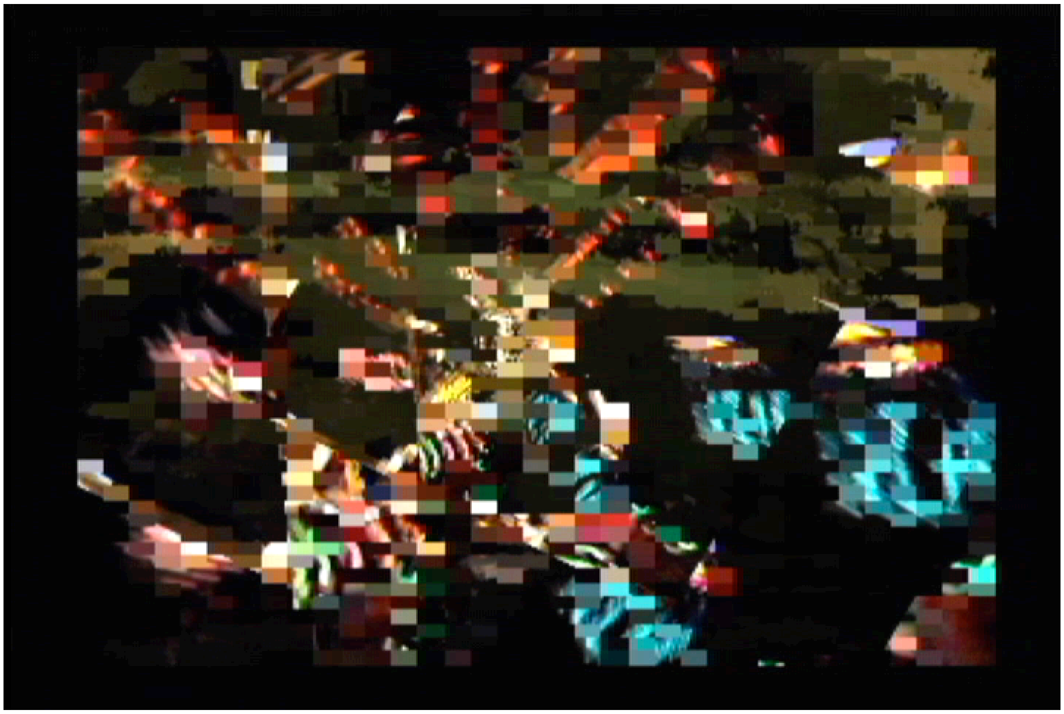


Fig. 36. Still image of jit.rubix filter, 2005

frequencies are jit.rubix, jit.sprinkle, and jit.plume. Each one of these filters fragments the image differently. The jit.rubix filter divides the original image into a user defined number of squares causing the image to have a pixilated effect without losing the integrity and resemblance of the original image (see Figure 36). Jit.sprinkle introduces spatial noise to the image, causing the image to look blurry and made of tiny dots (see Figure 37). The most animated and complicated filter is jit.plume, which displaces points based on luminance (see Figure 38). After defining the amount of squares into which the image is broken, each square is animated and pushed around the screen according to the amount of luminance of the entire image.

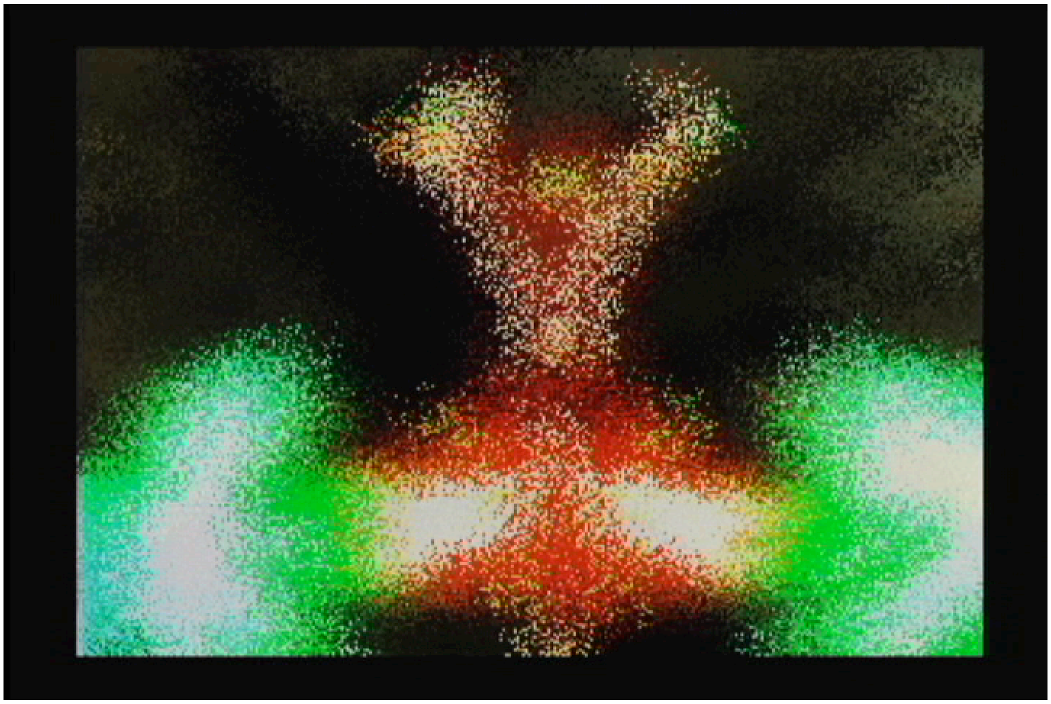


Fig. 37. Still image of jit.sprinkle filter, 2005

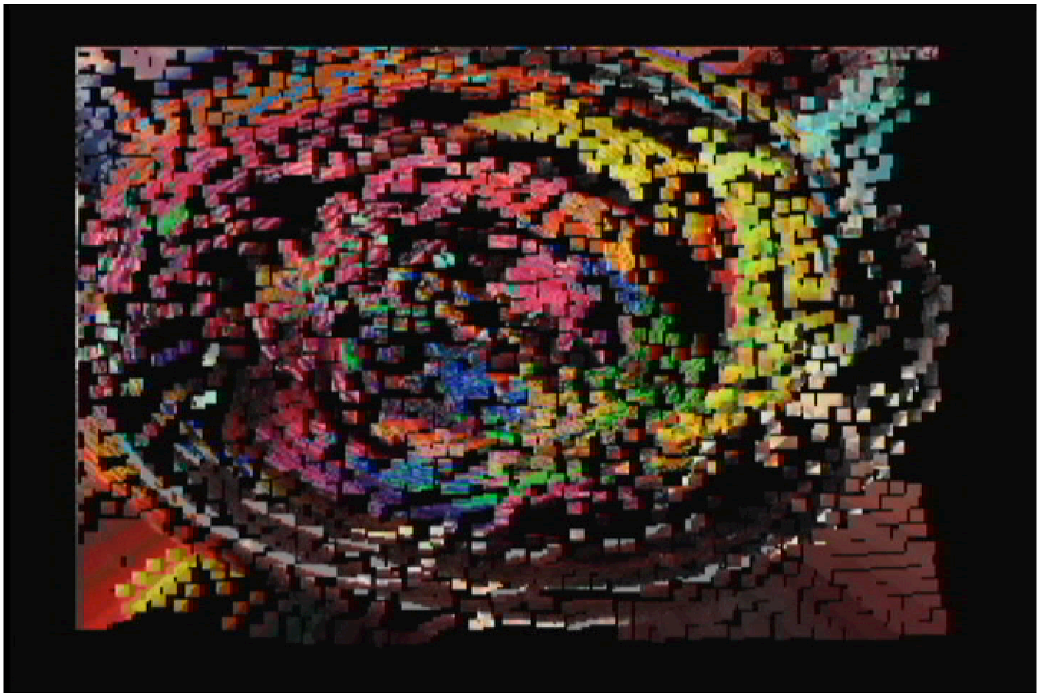


Fig. 38. Still image of jit.plume filter, 2005

These filters are incorporated into the program and activated when the high frequency sounds are produced. To accentuate the low frequency drum attacks that occur in the first section of the music, the filters jit.scanline, jit.plur, and jit.repos are implemented. Jit.scanline uses a one-dimensional matrix to offset the scanline of the image creating an overall sinewave distortion pattern (see Figure 39). Another filter, jit.plur, produces an image composed of many small squares which inherits the colors and subtle shape of the original image, yet is distorted enough to become disconnected from the original image (see Figure 40). Jit.repos repositions the pixels to create a bulge in the image (see Figure 41). Each of these filters creates a unique connection between the image and sound.

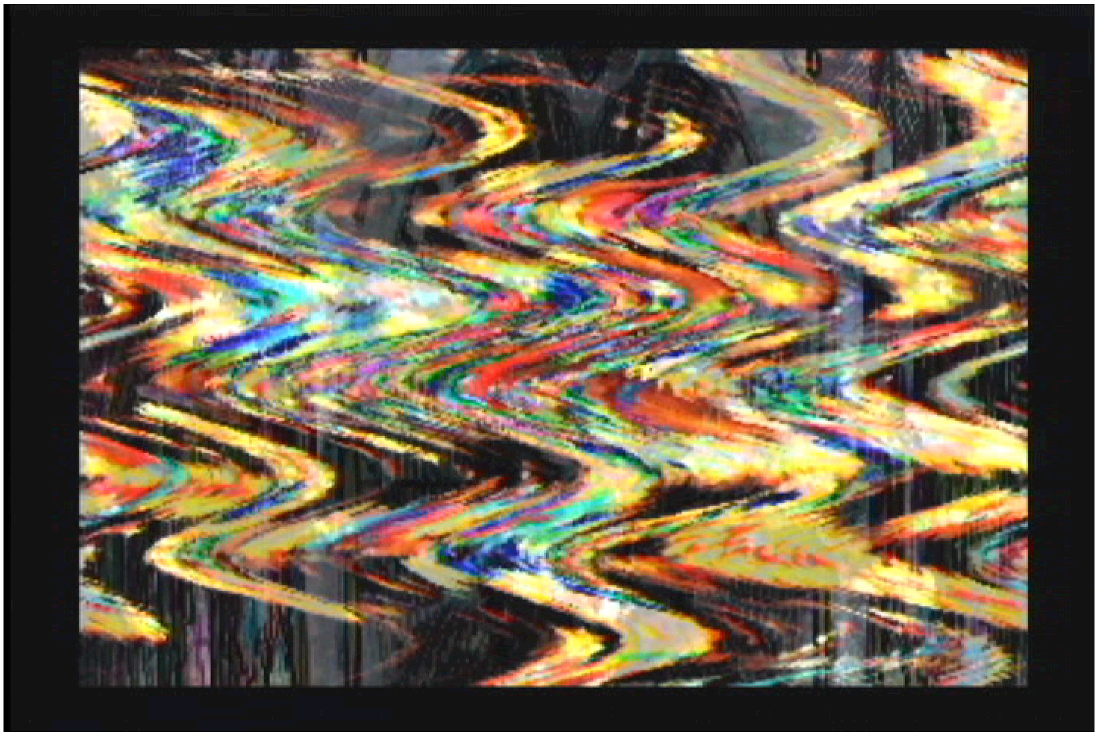


Fig. 39. Still image of jit.scanline filter, 2005

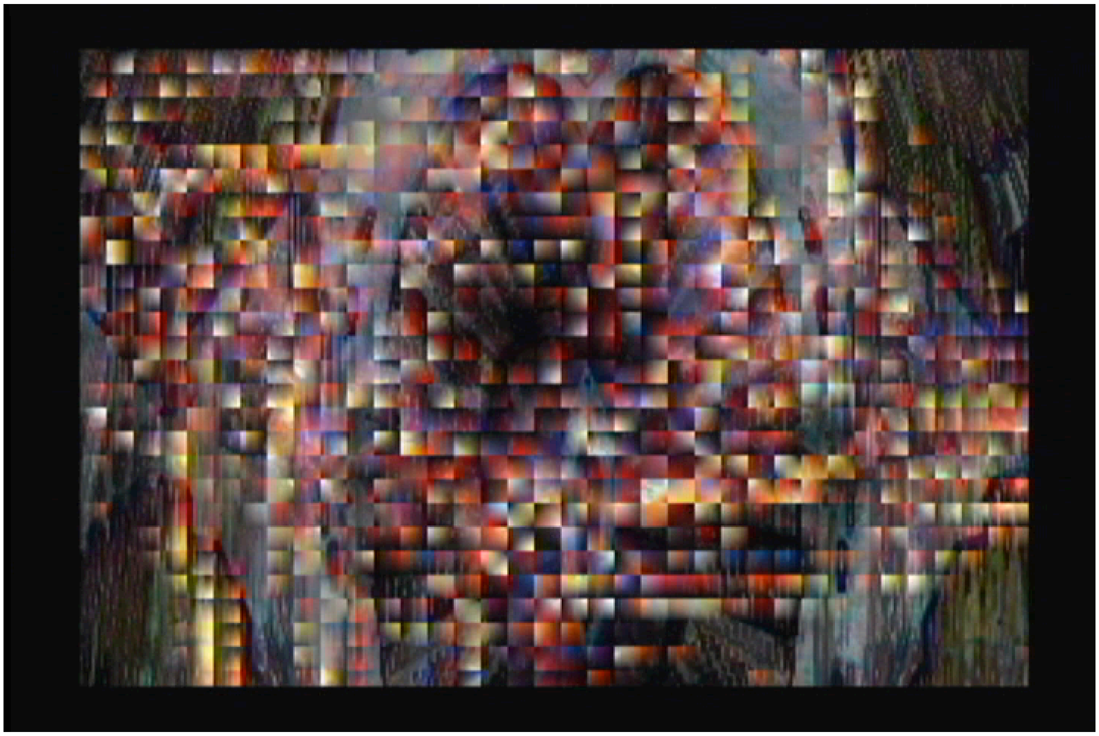


Fig. 40. Still image of jit.plur filter, 2005

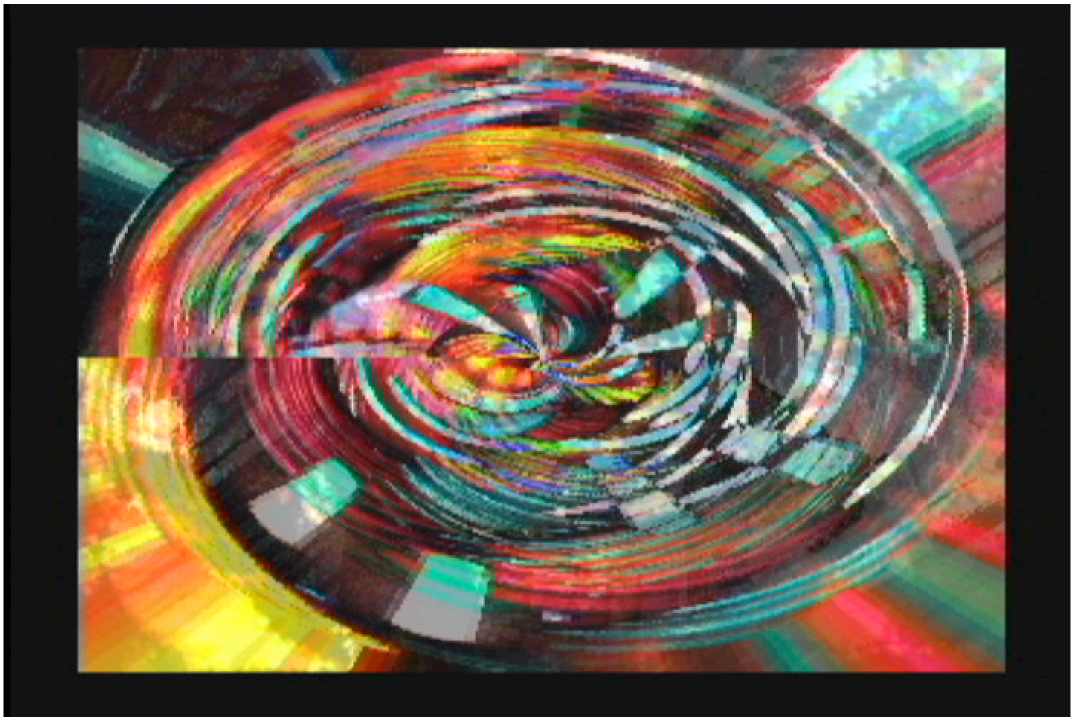


Fig. 41. Still image of jit.repos filter, 2005

Since these filters are produced in real time and require large amounts of processing power, it is important to efficiently integrate them into the patch by allowing the artist to turn them on and off during the show. This means only the filter being used is turned on, and the others remain off until needed. The method of exposing these filters at the appropriate time by musical triggers follows the same guidelines as the other editing techniques within the patch. For example, the main video is fed into the High Filtering section containing the three filters. This video imagery is connected to only one filter at a time. User-defined hotkeys transfer the video to the corresponding filter and turn off the remaining filters. Therefore, one filter is always affecting the video during the performance, however it is only output to the screen when the instrument triggers its response. This ability to hide the filter until the desired moment is possible by using the Gate object. The Gate object has two inputs. One input is the unfiltered main video file and the other is the filtered video imagery. Gate allows the unfiltered imagery to pass through by default. When a high frequency instrument is played, the Gate object switches to allow the filtered imagery through. This method will reveal the filter at the appropriate time and give the impression of instantaneous filter response to the music. This gate switching method is applied to all video-filtering systems within the program.

IV.5. Working with the Orchestra

Once the form of imagery presentation has been resolved, it is important to test the program's functionality with live instruments before the actual performance. The orchestra only rehearses the selected song four to five times before the final performance. Testing the patch during these sessions for accurate audio response is necessary to make any unforeseen corrections to the patch. This form of live testing requires implementing live audio input; this process is discussed in the next section, "Maximizing Live Performance Variables."

These patch tests are important in determining what instruments to isolate with the microphones. When listening to the pre-recorded music, it is easy to assume what instruments should be accentuated. However, after building the patch, realizing the capabilities of the audio-filtering system, and testing with live audio input, it was determined that some instruments could not be isolated easily and another instrument was substituted for it. This recursive testing process helps determine the most responsive and effective audio input.

Only certain sections of the orchestra attend each rehearsal making it difficult to get an accurate measurement of all instruments at the same time. To compensate for this, live recordings of the instruments were collected for each section that is monitored by the microphones. These recordings are inserted into the program to simulate the live audio input. The filtering system is then adjusted to this audio as if it were the live audio input. This technique helps prepare the various thresholds for the live performance. Some adjustments must still be made to the filtering system and microphone input levels once the entire orchestra is playing together at the final performance.

IV.6. Maximizing Live Performance Variables

There are many variables within a live performance that cannot be predetermined, but must be taken into consideration when building the program. The artist must have control of these variables during the performance and be able to make adjustments in real time. Failsafe features are built into the program to allow for quick adjustments to any program function and compensate for these unforeseen variables. Flexibility within these controls allows the artist to generate effective audio-visual correlation and overcome any obstacle during the performance.

One live performance variable to pay attention to is the lack of synchronization between the pre-edited imagery and the pacing and of the performance. The symphony orchestra has the potential to play the music slower or faster than the recording file used during development, therefore shifting the intended audio-visual correlation. It is important to keep the intended correlation, because the visuals develop as the music develops. The solution is identifying the three main sections within the full-length video edits, which correspond to the three musical sections, and looping each video section. This is accomplished by using the function called looppoints to define each section within the entire video edit. For instance, the first musical section lasts for approximately two minutes in the recording, and the corresponding video plays for this amount of time. The main video edit is set to loop between the first frame and last frame of this particular section while the orchestra is playing the corresponding music. If the orchestra plays this section longer than the pre-determined two minutes, the video will begin to loop within the designated looping points. This will avoid any dramatic change in visuals that would occur if the video plays into the next section before the audio of the current section is complete. At each section, the artist presses a key to activate the corresponding looping points and begin to play only that section of video. Each section is designed to play to the end of its looping points and then reverse and play the footage backwards. Since the video for each section progresses with the music from subdued motion and contrast to extreme intensity, this will ensure that the video is still at an intense point if it starts to loop and matches the music.

When using live musical input instead of the recording, several adjustments must be made to the filtering system. The first adjustment is to capture the live audio and incorporate it into the program. To detect the sounds of specific instruments, microphones are strategically placed within the orchestra. Each microphone is then plugged into an audio interface called the Traveler, manufactured by Mark of the Unicorn (MOTU) [9], which allows for multiple audio signals to be simultaneously sent to the computer via FireWire cable. After changing the software preferences to receive the live microphone input, the audio filtering section of the program is adjusted to direct each microphone input to a its own svf~ filtering object. Each svf~ object employs a specific filtering method according to the instrument it is monitoring. For instance, the microphone attached to the flute sends the audio signal to its designated high-pass filter object. Any low frequencies beyond the cutoff threshold are eliminated from the signal. Each instrument that is monitored requires different cutoff frequencies than were determined with the recording. Using the audio interface to adjust the volume levels of the microphone before it is transmitted to the computer program assists in highlighting the instrument sound from the rest of the orchestra. Combining the use of the Traveler for external audio adjustments with the right cutoff frequencies within the program generates optimum emphasis of selected instruments.

Another problem that can occur during a live performance is too much activity from one instrument and is avoided by disconnecting the audio source within the computer program. For instance, the flute might have moments during the music where it is very prominent and should be accentuated by the visuals, yet other moments where it blends with the rest of the orchestra, playing constantly, and should not be visually showcased. In this case, the flute triggers the distortion filters, which can become too overwhelming and ineffective if these filters are constantly shown. To solve this problem, the ability to connect and disconnect each microphone input from its triggering mechanism is implemented with corresponding keystrokes. During the performance, for example, this program enables the artist to press the number four key disconnecting the flute from its filtering system in real time. The instrument can then blend with the symphony and play as needed, yet not distract the viewers by triggering unwanted visual responses.

IV.7. Recursive Testing and Patch Refinement

Once the video is efficiently integrated into the patch and the proper interactive elements are established, a recursive process begins of testing the patch with the music to determine its effectiveness in the performance. How the visual information is revealed through the patch changes as the visualization process develops. Since efficient use of the computer's processor is imperative to a successful show, the organization and use of functions within the patch are tested and reworked as necessary. Some functions may be eliminated or exchanged, and the interactive signals rerouted. Replaying the patch using the recording and evaluating the correlation of imagery with the music helps to determine necessary visual changes. When the imagery does not emphasize a selected moment in the music, it is replaced by another clip or further processed to match the desired style. Throughout this visual refinement stage, new styles of processing the footage can be developed to enhance moments in the music. The new video footage is again tested with the patch to ensure audio enhancement.

During this development process, peers performed an ongoing evaluation of the program and its functionality. All participants involved in the performance, which includes three other visual artists, the conductor/director and members of the orchestra, provide feedback on the aesthetic and technical aspects of the visual presentation. The conductor provides important advice on how to establish the ideal visual tone and tempo of the presentation to correlate with his conducting style. By gathering opinions from collaborators of various backgrounds, a well-rounded final piece will be formed. This type of multi-perspective opinion helps keep the overall performance in balance and entertain a wider audience.

CHAPTER V

RESULTS AND EVALUATION

V.1. Evaluation of Results

The success of this project was evident with the accomplishment of certain goals. The first goal was creating a program able to incorporate all necessary tools to allow for significant flexibility during the live performance. Not only did the program produce audio-responsive visuals in real time, but the operator was able to adjust all instrument inputs, and filtering options as needed during the performance. The second goal required that the program handle any changes in tempo, length of performance or human error in timing. By implementing the video looping technique, the appropriate visuals were displayed at the desired moment. The use of live musical input to trigger a real-time visual response ensured accurate synchronization. The third goal was for the program to be as efficient as possible to allow for precision response of images to audio. This was accomplished by refining the patch to effectively utilize the computer's processor to minimize a lag in response. Since these requirements were met, the visuals responded to the audio as soon as the audience heard it and created a seamless unity of image and sound.

Producing an entertaining, visually interesting and music-enhancing performance was another goal accomplished. A method to determine this success was subjective evaluation from both the audience and the artist. Observing the audience's response after the performance suggested the level of entertainment experienced. The amount of excitement generated was apparent by the applause and comments after the show. Several audience members with and without artistic backgrounds complimented the performance and inquired about the process to create such a show. Another goal to determine success was if the designer was satisfied with the visual performance. Success was undeniable as the artist felt a personal satisfaction that the performance yielded harmonious image and sound with interesting imagery. Public excitement about the project could lead to future performances in other venues. These performances would be another sign of success, proving its appeal to different audiences. This is an ongoing process, which has so far yielded two possible venues for future performances.

This performance proved to be an overall success, yielding only a few issues with the program and its implementation. One issue was discovered in the looping process, which created a repetition of images. Once the video section began to play backwards the images that were just seen for the first time were being replayed immediately. This duplication of imagery could be monotonous to the viewer and therefore distracting to the overall visual performance. It was important to keep the viewer always guessing what would happen next, and this looping method diluted the concept of mystery. A solution for this problem would have been to create an edit for each section that was much longer than what was thought necessary. This way, if the orchestra played longer than supposed to, the visual section would never have to loop.

Another issue was the processing speed of the computer. Even though the computer used in this performance, a PowerBookG4 with1.67 GHz processor and 2GB RAM, was sufficient to achieve all goals, the program's functionality could have been extended with a more powerful computer. Some of the editing techniques used to help efficiently process the imagery could have been expanded upon, revealing more options for visual presentation. Additional filtering systems could have been implemented to add even more variety to the project. The ability to create a more efficient and visually interesting patch would be possible with great computing power.

This project is a guide for other artists in attempts to unite visuals and music for a live musical performance.

CHAPTER VI

CONCLUSION AND FUTURE WORK

VI.1. Conclusion

Creating a visually entertaining performance that enhances live music through real-time audio-responsive visuals requires a recursive process of implementation, evaluation and correction. The methods executed in this thesis have proven successful. Once the music is determined, selecting the appropriate footage, recording it using various framing methods, and abstracting the footage to produce non-representational imagery is the first step in this visual performance. Integrating this footage into a custom, interactive program to display the video images in real time is the next step. It is important during this procedure to develop methods that efficiently process the audio signals and output the video. Recursive testing of the functionality of the patch and appropriate use of video footage will ensure precise audio-visual synchronization and visual enhancement.

VI.2. Implications for Future Work

The techniques presented in this paper are just a few examples of what can be accomplished with this type of interactive audio-visual performance. Artists who continue to develop presentations of interactive visual music can explore options of projection surfaces, methods of projecting images, types of interaction/response between the software and participants, creating responsive non-video imagery, and presentation in other venues.

Artists can manipulate the audience's viewing experience by altering the projection surfaces and breaking away from the boundaries of the stage. Projection surfaces confined to a flat, motionless rectangle become moving organic shapes that can add to the complexity of the imagery. Using multiple projections to surround the audience and extend the image beyond the stage invites the audience to feel like it is a part of the performance.

There are various ways to increase the interaction between the performers and the imagery. Sensors such as accelerometers, which measure acceleration, gravity or speed, and flex sensors, which detects angles, can be attached to the musicians or conductor to employ their motions as driving forces for the imagery. In contrast, the performers could change their performance in response to the images being projected, causing the imagery to drive the musicians. Another way to involve the participants is by setting up a video camera that captures the live performance and simultaneously uses it as imagery for the program, integrating the video image into the final edit in real time.

Although this proposed performance uses strictly video images, there are several other ways to visually enhance the music not involving time-based media. Audio-interactive sculptures that respond to the sound have the potential to create interesting visual movement. Choreographing lights to highlight particular tones within the music can generate an effective ambiance. Incorporating physical environmental effects such as smoke, moving air or even water could affect the participant in a deeper level by engaging the sense of touch. The possibilities of incorporating responsive non-video media with music are only limited to the imagination and resources of the artist.

This responsive visual project is not limited to an orchestra performance, but can be implemented with smaller musical groups or at amusement parks, museums, and entertainment venues. By adjusting the imagery accordingly, audio-responsive visuals will enhance the music performed by various sized ensembles playing different types of music. To enhance the mood at various entertainment venues, panels of responsive imagery could surround the viewer and respond to selected sounds within the environment. These sounds may vary from music played at the venue, crowd noises, or ambient ride noises. An option to expand the connection between participants and their environment is to have the visuals respond to audio that is in a separate and distant location from the panels. This setup will help unite the entire venue. Instead of responding to audio, the images could respond to pressure sensors set in the floor that the crowd triggers as it moves around the space. There are many types of interaction that would add to the excitement of the environment.

Exploring these options can lead to unique and exciting experiences for participants. It is important to continue exploring new options and breaking the boundaries of audio-visual entertainment to further develop and mold the evolution of artistic entertainment.

REFERENCES

[1] M. Chion, *Audio-vision: Sound on Screen*, Editor and Translator Claudia Gorbman. New York: Columbia University Press, 1994.

[2] Cycling ’74, “Cycling ’74 Product Overview.” http://www.cycling74.com, 2006.

[3] N. Currie, "VJ Culture: Design Takes Center Stage." Adobe Systems, Inc. http://www.adobe.com/designcenter/thinktank/vjculture/, 2006.

[4] E. Epstein, *“Untitled”, Opus 161, 1965/66*, Courtesy of http://www.lumia-wilfred.org, 2007.

[5] E. Fischinger, “Writing Light.” http://www.iotacenter.org/Elfriede/WritingLight.htm, 1998.

[6] G. Levin, *Painterly Interfaces for Audiovisual Performance*. Master’s Thesis. Cambridge, MA: Massachusetts Institute of Technology, 2000.

[7] M. Lew, “Live Cinema: Designing an Instrument for Cinema Editing as a Live Performance”. *Proceedings of NIME 2004*, 2004, pp. 144-149.

[8] B. Likins, "Sun Rings." *Kronos Quartet*. http://kronosquartet.org/SR/prog3.html, 2004.

[9] Mark of the Unicorn, *Traveler* FireWire audio interface, http://www.motu.com/products/motuaudio/traveler. 2005.

[10] W. Moritz, “The Dream of Color Music, and Machines that Made it Possible,” *Animation World Magazine*, Issue 2.1, April 1997. http://www.awn.com/mag/issue2.1/articles/moritz2.1.html.

[11] L. Pocock-Williams, "Toward the Automatic Generation of Visual Music." *Leonardo,*

25 (1992), pp. 29-36.

[12] F. Popper, *Origins and Developments in Kinetic Art*. Greenwich, CT: New York Graphic Society, 1968, pp. 156-172.

[13] C. Roads, *The Computer Music Tutorial.* Cambridge, MA: MIT Press, 1996.

[14] R. Rowe, and E. L. Singer, "Two Highly Integrated Real-Time Music and Graphics Performance Systems." *Proceedings of ICMC 1997*, 1997, pp. 269-272.

[15] R. Rowe, *Interactive Music Systems: Machine Listening and Composing*, Cambridge, MA: MIT Press, 1993.

[16] J. Shepter, "Sun Rings." Design Interact. http://www.designinteract.com/features_d/sunrings/. 2006.

[17] T. Wilfred, "Composing in the Art of Lumia," *Journal of Aesthetics and Art Criticism*, (VII) December 1948, 79-93.

[18] T. Wilfred, "Light and the Artist," *Journal of Aesthetics and Art Criticism*, (V) June 1947, 247-255.

[19] W. Williams, "Sun Rings: Complete Program." Kronos Quartet. http://kronosquartet.org/SR/prog1.html. Aug, 2002.

VITA

Allison Brooke Beane

Visualization Laboratory
Texas A&M University
C418 Langford Center
3137 TAMU
College Station, TX 77843

bbeane@viz.tamu.edu

Education

M.S., Visualization Sciences, Texas A&M University, 2007

B.E.D., Bachelor of Environmental Design, Texas A&M University, 2002