A Nossa Universidade

Colégio dos Jesuítas
Rua dos Ferreiros - 9000-082, Funchal

Tel: +351 291 209400
Fax: +351 291 209410
Email: gabinetedareitoria@uma.pt

DM

Interactive Music Visualization – Implementation, Realization and Evaluation

Marco Filipe Ganança Vieira

UNIVERSIDADE da MADEIRA

DM

Interactive Music Visualization – Implementation, Realization and Evaluation

MASTER DISSERTATION

Marco Filipe Ganança Vieira
MASTER IN INFORMATICS ENGINEERING

UNIVERSIDADE da MADEIRA
A Nossa Universidade
www.uma.pt

September | 2012

UMa | Nome do Projecto/Relatório/Dissertação de Mestrado e/ou Tese de Doutoramento | Nome do Autor

DIMENSÕES: 45 X 29,7 cm
PAPEL: COUCHÊ MATE 350 GRAMAS
IMPRESSÃO: 4 CORES (CMYK)
ACABAMENTO: LAMINAÇÃO MATE

NOTA*
Caso a lombada tenha um tamanho inferior a 2 cm de largura, o logótipo institucional da UMa terá de rodar 90º , para que não perca a sua legibilidade|identidade.

Caso a lombada tenha menos de 1,5 cm até 0,7 cm de largura o laoyut da mesma passa a ser aquele que consta no lado direito da folha.

UNIVERSIDADE da MADEIRA

UNIVERSIDADE da MADEIRA

Interactive Music Visualization – Implementation, Realization and Evaluation

MASTER DISSERTATION

Marco Filipe Ganança Vieira
MASTER IN INFORMATICS ENGINEERING

ORIENTAÇÃO
Mon-Chu Chen

Abstract

This thesis describes all process of the development of music visualization, starting with the implementation, followed by realization and then evaluation.

The main goal is to have to knowledge of how the audience live performance experience can be enhanced through music visualization. With music visualization is possible to give a better understanding about the music feelings constructing an intensive atmosphere in the live music performance, which enhances the connection between the live music and the audience through visuals. These visuals have to be related to the live music, furthermore has to quickly respond to live music changes and introduce novelty into the visuals. The mapping between music and visuals is the focus of this project, in order to improve the relationship between the live performance and the spectators. The implementation of music visualization is based on the translation of music into graphic visualizations, therefore at the beginning the project was based on the existent works. Later on, it was decided to introduce new ways of conveying music into visuals.

Several attempts were made in order to discover the most efficient mapping between music and visualization so people can fully connect with the performance. Throughout this project, those attempts resulted in several music visualizations created for four live music performances, afterwards it was produced an online survey to evaluate those live performances with music visualization.

In the end, all conclusions are presented based on the results of the online survey, and also is explained which music elements should be depicted in the visuals, plus how those visuals should respond to the selected music elements.

Keywords: Music Visualization; Computer Graphics; Audio-visual Systems; Acoustic Application.

Abstracto

Esta tese descreve todo o processo de desenvolvimento de visualização musical, iniciando com a implementação, seguido pela realização e posteriormente avaliação.

O objectivo principal é obter o conhecimento de como a experiência da audiência de uma performance ao vivo pode ser melhorada através de visualização musical. Com visualização musical é possível dar uma melhor compreensão sobre os sentimentos da música construindo uma atmosfera intensa na performance musical ao vivo, a qual aperfeiçoa a conecção entre a música ao vivo e a audiência através dos visuais. Estes visuais têm que estar relacionados com a música ao vivo, para além disso têm que responder rapidamente às mudanças da música e introduzir novidade aos visuais. O mapeamento entre a música e os visuais é o foco deste projecto, podendo assim melhorar a relação entre a performance ao vivo e os espectadores. A implementação de visualização musical é baseada na tradução de música para visualizações gráficas, portanto no início o projecto foi baseado em trabalhos existentes. Mais tarde, foi decidido introduzir novas formas de transmitir música pelos visuais.

Várias tentativas foram realizadas para poder descobrir o mapeamento mais eficiente entre música e visualização para que as pessoas possam conectar-se totalmente à performance. Durante este projecto, estas tentativas resultaram em várias visualizações musicais criadas para quatro performances musicais, depois foi produzido um questionário online para avaliar essas performances ao vivo com visualizações musicais.

No fim, todas as conclusões são apresentadas baseadas nos resultados do questionário online, também é explicado quais são os elementos musicais que devem ser retratados nos visuais, e ainda como esses visuais devem responder aos elementos musicais selecionados.

Palavras-Chave: Visualização Musical; Computação Gráfica; Sistemas Audiovisuais; Aplicação Acústica.

Acknowledgments

In the first place I want to express my gratitude and appreciation for my thesis advisor, Dr. Mon-Chu Chen, for all the teaching, support and patience throughout the development of this thesis.

Secondly, I would also like to thank my family, specially my parents, godmother, and grandmother for all care, understanding and support during this period of my masters' degree.

Finally, I have to be thankful for all help and advices given by my friends and colleagues, as well as all the mandolin orchestra elements and my mandolin teachers, because they helped me become more sensitive and passionate about music. This is why I wanted to do this thesis, because it is the combination of music and computer science, my two life passions.

Table of Contents

Figures

Tables

1. Introduction

Music is a unique form of expression due to all its special characteristics. These characteristics are what make music so special, because they are constantly changing, growing with different levels of intensity as it develops over time. This artistic form of expression can also inspire emotions, in other words music composers/ musicians can express their feelings through music, which then becomes a source of inspiration for other musicians or even people who carefully listens to music. Additionally, those special characteristics mentioned before, are the several elements existent in the music structure itself, which is understood only by proficiency people in music (Klemenc, Ciuha, Subelj, 2011). The music structure elements is what makes it so intensive and emotional, this means that music artists use those elements in order to express their feelings through their music. Music is so interesting because is constantly changing and growing over time and has different levels of intensity as it develops due to the previous stated elements.

When people listen to music, they sometimes create a mental model of the specific music piece that they are listening to. That mental model represents the message and/or feelings of the person at that moment of listening the piece, which sometimes are highly related to their personal life experience. These mental models can be different for each person, because an individual has their way of seeing and feeling things. This is why a music piece sometimes has a vast number of unique interpretations for the same music piece. Everyone has their own life experiences and influences, as well as culture, therefore, everybody have different opinion when listening to a music piece.

A live performance exalts different characteristics of music, because they become so evident as the audience can experience different levels of emotional intensity existent in a live performance. In consequence the audience navigates through a parallel world produced by inherent emotions resulting of the different music pieces. Moreover, with a music live performance, there is the possibility of providing a series of stimulating reasons, which provokes significant moments intrinsic to the listener's live experiences.

However, not all the audience have that emotive availability and sensorial (for example deaf people, or even people emotionally closed). These people need attractive and complementary stimulation in order to help them with those "hidden" feelings and emotions in music. Thus, we can considerate music visualization as an extrinsic emotion guider to the musical moment which they are living. With technology is possible to build this extrinsic emotion guider to the musical moment, in order to make the connection between live music and the mental world created by the audience. Visualizations in a live music concert should expand, enhance and re-represent music elements such as volume, pitch, timbre, etc. in real-time. These elements can be conveyed in visuals, creating a relationship between the live music and the visuals. The computer "listens" the music and then triggers different visuals reacting accordingly to what it is listening at the moment, thus people can "see" music (Hiraga, Watanabe, Fujishiro, 2002), which is good for those people who can't hear. It helps them giving a perception and comprehension of what kind of music it is through the visualization.

These elements can illustrate how a composer developed his or her ideas in a piece of music, and how they are important elements that should be considered when listening to music. In a piece of music, the audience should be able to understand how integral thematic elements are to a music composition, but how they influence the visual design process of music visualization? Can themes in music successfully transfer to the visual animation? The work documented in this thesis explores some of the possibilities by combining computer animation as a visual extension to a live music performance.

Musical and visual explorations expand the boundaries of a computer as a content creation tool to elevate the computer as a visual instrument for the artist for the execution of the real live graphic animations. During the live performance, the system unites the computer and the artist. The computer becomes a visual instrument for the artist to play and its visualizations reacts to live music performance (Bain, 2008).

1.1. Goals and Motivation

The objective of this project is to find the best mapping between music and visualization. This connection between music and visuals is created by the union of technology and music. This is a very interesting merger which may improve the spectators live music performance experience. This It is believed, that if the audience of a live performance with music visualization can understand music elements through the visuals, then they can have a better understanding of the feelings of the music piece and enhance their live performance experience.

Music is comprised of a very complex structure, which only skilled people in music can understand them. But music visualization tries to give a chance for everybody to understand it, specially the feelings/message behind each music piece expressed through music elements (pitch, tonality, dynamics, etc.). It is also believed, that music visualization enhances the live performance experience of the audience, as makes it more intensive and improves the connection between people and the music using music visualization. People become more involved with the music they are listening to at the moment with music visualization because it helps them to create a mental model of the music, thus they understand it better and also make it more interesting. This is the main goal for this project, enhance the audience live performance experience using music visualization. They should become emotionally more involved with music. Music visualization is that complement for those people who struggle to understand a certain music piece or have difficulty to connect to a certain kind of music, making them feeling or imagining something when listening to it.

The basis for the achievement of this objective is having a good mapping between music elements (pitch, volume, tonality, etc.) and the visual. Moreover, the design of the visuals should be really good and consistent with the particular music piece.Throughout the development of this project, will be tested several mapping functions in order to know what are the best options and in the end, there will be a survey so we can know the feedback from the project and evaluate it.

1.2. Contribution

The main contribution of this project is the mapping function between music elements and visualizations. In this project, it will be tested several mappings such as: pitch – location, pitch – color, etc. According to all research, all mapping functions used were never evaluated and proved that they were really effective for music visualization. This is the reason why it is decided to try different approaches to depict music through visuals and then make an evaluation in order to know if the best way to convey music through visualizations.

This mapping function between music and visuals is important, because as it was explained before, musicians express their feelings through music. And using music elements they can express any kind of feeling they want to. These elements is what makes each music piece unique with their own feelings, so the visuals should be consistent and show these music elements in a way that people can feel the same when they listen the music without visuals. If there is no mapping between music and the graphic animations, then the graphic animation might not be coherent with music, resulting in a visual unfaithful to the message of certain music.

There were several attempts in the past in order to find the best relationship between music and visualization. There are some interesting related works that shows several approaches for this problem that will be explained in detail in section 2.1.2. This thesis will try to discover the best approach in this field of music computing.

1.3. Synopsis

The thesis structure is organized in seven sections. In this section will be explained how each section is organized and what contains. The first one is the introduction of the theme music visualization, and what is envisioned to do in this project.

The second section is the literature review, where is explained and mentioned all related work. All bibliography used for this thesis is described in this section, besides that is also explained why they are important for this project, as well as their contribution for this thesis. The consulted references are divided in Music Visualization (section 2.1), Available Technologies (section 2.2), and Quartz Composer (2.3).

Then in section three, is defended the music elements decided to use for the mapping between music and visualization. It is explicated the music elements such as tonality, volume, so the reader without music feedback can understand everything that is described in this thesis.

In section four is explained the methodology for this project. As it was mentioned before, the methodology thought for this project is to find the best relationship between music and visuals therefore the audience in a live performance can evolve into a more intensive level of music perception highlighting its underlying feelings. So, in this section, it is developed the music mapping into visualization, and how it changed during this project. After that, the implementation is explicated in terms of software (section 4.2.1), hardware (section 4.2.2) and finally visualizations (section 4.2.3).

In section five is described the several live performances that occurred during the project, and what went wrong, what kind of problems emerged as well as solutions. These live performances were good for testing the mapping between music and visuals helping finding out the best method to map music into visualization. The level of complexity increases throughout the development of the project. It first started with a small presentation in the first semester with only one music (section 5.1), followed by two concerts at the Wine Institute with a mandolin orchestra (sections 5.2 and 5.3) which the second one was an improvement of the first, because all problems of the first concert at the Wine Institute were solved and the goal pretended for that live performance was achieved. Finally, the last challenge was the music visualizations for a musical drama piece of music undergraduate students project, which there was a great feedback (section 5.4).

In section six is presented the results from the online survey about all these music visualizations live performances, where is also presented charts with the intention of proving the hypotheses formulated in this section for this thematic. The results of the survey are discussed, and conclusions are also presented.

Lastly, it is explained all conclusions in detail of this thesis as well as the results. Plus, it is described the future work that can be developed in music visualization.

2. Literature Review

2.1. Music Visualization

In the current days, music is intertwined in almost everyone's life, mainly because it helps them to express themselves (mostly through the audible sense), but sometimes it is also possible to relate to music by visual means (Isaacson, 2005). Music visualization is a feature found in media player visualizers' software such as Windows Media Player, or iTunes, etc. This feature exhibits animated imagery based on music. The imagery normally is rendered in real time and synchronized with the music, which is being played.

Music has been combined with visualizations for a long time, especially in artistic areas, such as: Operas, ballets, musical dramas and movies (e.g.: Fantasia by *Walt Disney*). There is a better comprehension of the performance expression when the audience of a musical performance can visualize it (Hiraga, Watanabe, Fujishiro, 2002). The reason why is necessary this correlation between the auditory and visual senses, is due to the complicated structure of music for those who aren't specialist. Therefore, music visualization is a feature, which helps the audience understanding those complex music structures, such as the music feelings and purposes (Klemenc, Ciuha, Subelj, 2011). It's necessary to find characteristics of music with some analysis methods in order to know which information should be shown on the visualized figures for expressive performance (Hiraga, Mizaki, Fujishiro, 2002).

Visual performances provide observers an explanation and judgment for musical expression, it is a visual help to understand and analyze performances and their musical structures. Media mapping from sound to visual information leads the observers to synesthesia (Hiraga, Mizaki, Fujishiro, 2002). This means that, this visual information is the result of the blending of two senses, for instance, associating an image to a particular music. In this case, the visual sense is stimulated by the auditory sense, when somebody is listening that specific music, can relate it to a certain image for some reason.

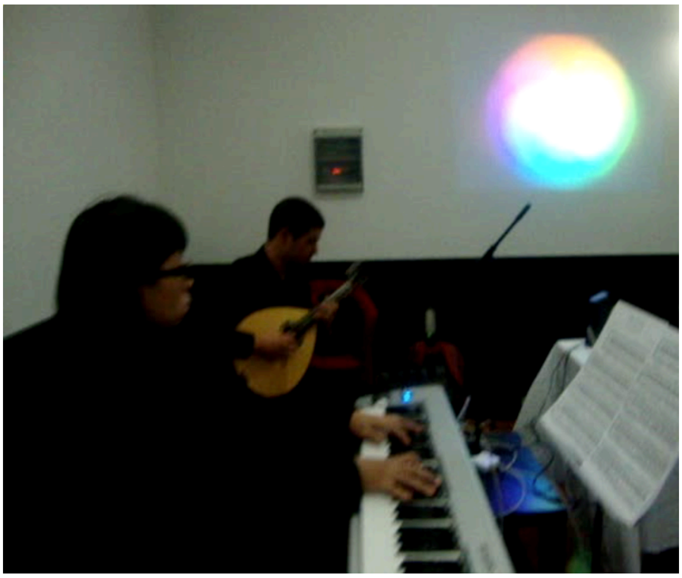


Figure 1 – First Live Performance with Music Visualization done by the author of this thesis

Music can be a “physical object”, which the audience can examine it and analyze it at any time, when the audience of a performance is able to "watch" music through pictures or images or some kind of visualizations (Isaacson, 2005), like in figure 1. In other words, when the audience can visualize, is easier to explore music than just listening. Then, the visualizations take people’s experience of viewing a performance to another dimension, where they have other idea and thought about the musical piece that they are listening. Those visualizations could be pictures, visual effects, it is something visual that is associated with the music and it responds accordingly to the music changes (mood, or exposed feelings). People should feel something when they listen to music, these visualizations, makes it simpler, and thus, people comprehend the implicit emotion of the music.

If the concept of musical structure is applied in conjunction with the musical theory, then the interactive visualization of the music can be an effective tool for the exploration of the musical information (Isaacson, 2005). This is a great achievement, because it helps to prove that music visualization enhances the audience comprehension of the music information retrieval. Only when the visualizations really help, hence bad correlated music visualizations confuses the audience understanding of the music. So, it should be studied techniques (based on musical theory) to guide designers in a way, that their visualizations are really correlated to the music. There are some inputs to consider for the visualizations: the music's volume and the frequency (or the music pitches). These inputs are used as parameters in the mapping functions of the visualizations that are being displayed in real-time. For instance, if the volume of the music is low, then the size of the objects in the visualization should be smaller, otherwise, should be bigger.

2.1.1. Objectives with performance visualization

With music visualization it is pretended to achieve several goals that are capable of enhancing the audience live performance experiences. Therefore, the main focus is the understanding of musical intention of performances (Hiraga, Mizaki, Fujishiro, 2002). The musical intention of performances can be understand by the mood of the music. Mood is an important characteristic to be presented in visual performances in order to give the desired atmosphere. The significance of these moods is how people can access and understand performance data that is considered as subjective (Hiraga, Mizaki, Fujishiro, 2002).

Moreover, people should be able to compare their thoughts of live music visualization performances against simple live music performances (without visualization) among them. Furthermore the sharing of the understanding of both kinds of performances should also involve the players (musicians) in order to have an opinion of a skilled person in music. In other words, sharing thoughts about this experience helps people to understand what they are feeling, and if these feelings are common to everybody or not. Thus, the process of understanding the music intentions might be easier and also helps improving the project in accordance with the audience criticism.

2.1.2. Related Work

There are a big number of projects about this theme. In this section, it is referred some of those projects. These existing projects helped making some decisions about this thesis, consequently the project is based on those previous works in order to take advantages of the work already done and studied, thus the goal of this thesis can be achieved with quality.

2.1.2.1. ANIMUS

This application enables the interaction between a live musician and a responsive virtual character, and it responds according to the music it "hears" (see figure 2). This character visualizes the perception and cognition of musical input by exhibiting responsive behavior expressed through animation.

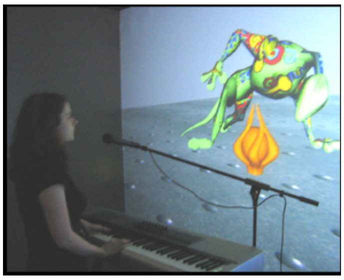


Figure 2 - User interacting with virtual character (Taylor, Boulanger, Torres, 2005)

This application is influenced by Cooke's work (Cooke, 1963), which talks about the feelings that are embedded in music. Those feelings are inherent to music, it isn't a technical part of music, but something more elusive. Cooke used western tonal music as reference for his work, and he makes hypotheses concerning the relationships between musical structures and composers' emotional intent. Through Cooke's work (Cooke, 1963), they decided to give a visual response according to the music emotions with a life-sized virtual character called ANIMUS. ANIMUS is capable of perceiving musical input and to encode it in a way consistent with previously defined organizational models for tonal music. Therefore, there is a cognition mechanism related to the musical-theoretical features to emotional states consistent with Cooke's findings, simulating an emotional understanding of the music it "hears" (Taylor, Boulanger, Torres, 2005).

The live musician sings into a microphone and there is also someone playing a digital piano. So, the first step is to extract meaningful features (pitch, amplitude, tone, chord), and finally these features must be organized in a consistent way with previous research in the area of human musical perception. Characters (ANIMUS) can assign happy feelings to a particular melody, which they find pleasing, in the other hand, they can assign a fearful feeling to a certain series of chords that they find threatening. This means that, the character should be flexible enough to stimulate an emotional response to music-theoretical features within a melody line. For instance, if a melody contains many instances of a minor third, than that melody is "tragic", but if there are many instances of a major third, then it's a joyful melody (Taylor, Boulanger, Torres, 2005).

2.1.2.2. Interactive Music Visualization

This application consists of three parts: sound analyzer; visualization mode; scene editor. The sound analyzer consists in an evaluation of some characteristics of each audio file input for 1/25 sec. such as: volume, balance (stereo position), dominant (solo), instrument position (melody), tempo (music rate), and music mood. Currently just volume and balance are implemented (Kubelka, 2000).

The Visualization mode offers two modes of rendering the objects from the scene editor (real-time mode and full-rendering mode). Those modes are: Real-time mode (The scene is rendered simultaneously with the playback of the sound record from de audio file; and Full-rendering mode (All animation is rendered frame by frame and resulted images are saved on a disc).

Finally, the scene editor is defined before the visualization starts, composed by specific author requirements and is changed according to the data created by the sound analyzer. It was used basic geometric objects (sphere, cube), which reacted with the music and their size, position, orientation and color would change. It was also used a particle system, and colored background (suitable for expressing lightning, dawn or dusk). The particle system (see figure 3) is good for the music visualization, because it is possible to express the dynamics (motion of particles) and the mood (through color) of the music (Kubelka, 2000).

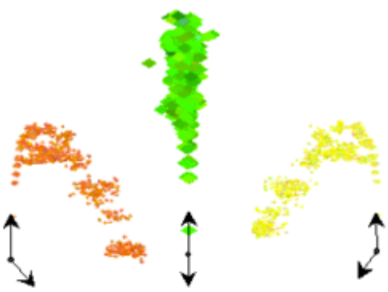


Figure 3 - System Particle with different parameters (Kubelka, 2000)

2.1.2.3. Comp-i

One of the first things to do in music visualization is to visualize the items of musical sense along music perception, for e.g., articulation, tempo, and dynamic change. They designed and implemented an application named "*comp-i*" to provide a 3D virtual space where users can perform visual information (Hiraga, Mizaki, Fujishiro, 2002). This application has three parameters, which are pitch, volume and tempo that are encoded as color saturation, diameter, and height of each cylinder (corresponds to a single note sound).

2.1.2.4. Visual and Aural- Visualization of Harmony in Music with Colour

In this project, they try to associate colors to musical tones instead of associating to individual pitches. Musical tones are comprised of three pitches with a certain interval between them. That different interval between pitches is what defines the music tonality. In figure 4, we can see the C Major tone, which is comprised by the three pitches C, E and G. Every tone is comprised by its specific three pitches (Klemenc, Ciuha, Subelj, 2011).

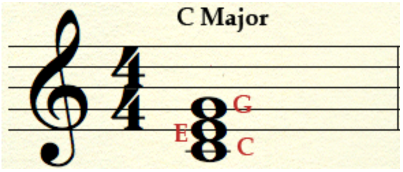


Figure 4 - C Major Tone

These tones can be major or minor, and normally, the major tones are associated to a more happy or moved musical piece (see figure 5), otherwise, it's related to a sad piece, which transmits sad or tragic feelings (see figure 6). So, for the major tones, they have saturated colors, in the other hand, for the minor tones, they have colors with low saturation (Klemenc, Ciuha, Subelj, 2011).

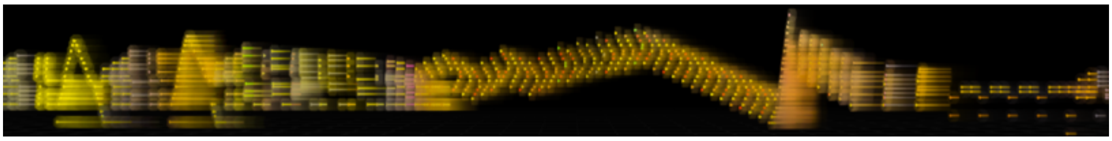


Figure 5 - Visualization Tchaikovsky's Walts of Flowers (Klemenc, Ciuha, Subelj, 2011)

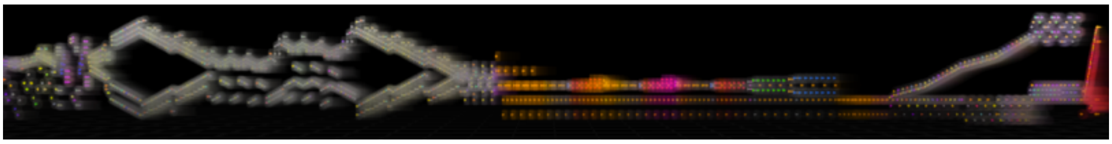


Figure 6 - Visualization of Prokofiev's Toccata in D minor (Klemenc, Ciuha, Subelj, 2011)

2.1.2.5. Emotion-based music visualization using photos

The goal of this project is to provide an emotion-based music player. This music player synchronizes photos with music based on its emotions. Several kinds of media (e.g.: music, photos) make people have different feelings, from sadness to happiness. The integration of these several media type can make a more touching presentation, as long as they are synchronized in emotions. By coordinating emotions in both auditory and visual contents, emotional expression in enhanced and user's listening experience is enriched (Chen, Weng, Jeng, Chuang, 2008). Based on theses facts, they decided to build a system to combine these two senses (visual and auditory). The inputs of their system are a song and a set of photos. The first step is extracting the emotion category from the music input and photos. They have eight categories: sublime, sad, touching, easy, light, happy, exciting and grand (see figure 7).

Figure 7 - Sample images of the different existing categories (Chen, Weng, Jeng, Chuang, 2008)

It is used Wu and Jeng's method (Wu, Jeng, 2008) for the music emotion detection. It consists using F-scores and multi-class classifier (built by SVM) to select the music features (Chen, Weng, Jeng, Chuang, 2008). For the photo emotion detection, they collected a set of several photos (398 photos), and each photo is tagged accordingly to its category, so they know what are the photos for each eight different feelings' categories. Bayesian classification method is proposed for automatic photo detection, in other words, they use Bayesian classifiers extracted and fed into classifiers to determine its emotion category (Chen, Weng, Jeng, Chuang, 2008).

2.1.2.6. MuSA.RT

MuSA.RT (Music on the Spiral Array Real-Time) is a system, which analyses music, and it's used for interactive music visualization through a real-time MIDI input, the live performance is processed, analyzed and mapped to a 3D model, revealing musical tonal structures such as pitches, chords, and keys (Chew, Franc, 2003). It is an interactive system for music analysis and visualization using the Spiral Array model (see figure 8). An expert listener is able to ascertain the keys and harmonic patterns when they listen to music. This process is difficult for novices, and the solution is providing visual cues such as geometric model (Chew, Franc, 2003).

This system provides a generic platform for testing and validating algorithms for real-time music tracking. A few actions permit the user to control better the system so he can have a better perception about the tonal structures. The user can zoom in and out, tilt the viewing angle and circle around the spiral to get a better view of the tonal structures and there is also an automatic pilot option, which is great to get the best view angle and center the camera at the heart of the action (Chew, Franc, 2003).

Figure 8 - MuSA.RT (Music on the Spiral Array Real-Time) system (Chew, Franc, 2003)

2.1.2.7. Visualizing Music

This project goal is to observe the audio structure of music. Lillie used a tool named *"The Echo Nest"* (The Echo Nest Corporation, 2012), which takes as input mp3 files and break it up into little segments, and saves the pitch, loudness (or volume) and high-level timbral information of those segments. Their program maps the audio data onto a visual scale and creates a video playback of the song (Lillie, 2008).

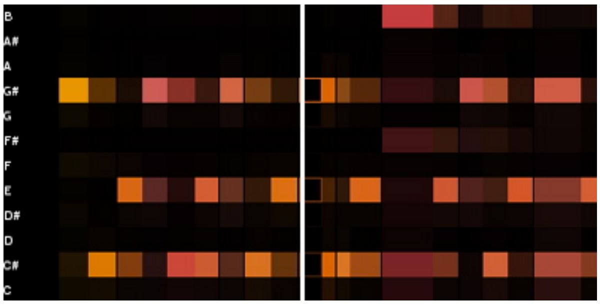


Figure 9 - Lillie's music visualization program (Lillie, 2008)

As we can see in the figure above, time is mapped onto the horizontal axis and the music is chopped into a series of segments (about 0.1 to 0.3 seconds long), which corresponds to the vertical slices that we see scrolling in the video playback. Each of those chopped segments is broken up into twelve pitches along the vertical axis (they are labeled with the corresponding note on a scale). The vertical slice is colored with an RGB value corresponding to the timbre of the sound in that particular segment. Timbre is the different sound of instruments (for instance, is what makes the mandolin sound different than a violin sound). Therefore, similar sound appears with similar color, otherwise the color is different (Lillie, 2008).

2.1.2.8. Visualization of Concurrent Tones in Music with Colours

Their aim is to visualize music by interconnecting similar aspects in music and in visual perception. Ciuha, et. al focused on visualizing harmonic relationships between tones and colors. In other words, instead of attributing a color to each pitch, they attribute color to tones. Those similar tones, chords and keys will be represented by a color with similar color hue, dissonance is represented by low color saturation (Ciuha, Klemenc, Solina, 2010).

In the following figure, it is possible to see the attribution of colors to the tones. As it was said before, musical tones are comprised of a sequence of three pitches (Ciuha, Klemenc, Solina, 2010).

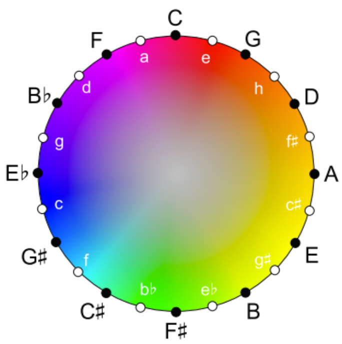


Figure 10 - Color association to musical tones (Ciuha, Klemenc, Solina, 2010)

The following figures (figure 11 and figure 12) illustrates the result of this system, where it is possible to observe the colors association to the musical tones.

Figure 11 - Excerpt from Pachelbel's Canon in D Major (Ciuha, Klemenc, Solina, 2010)

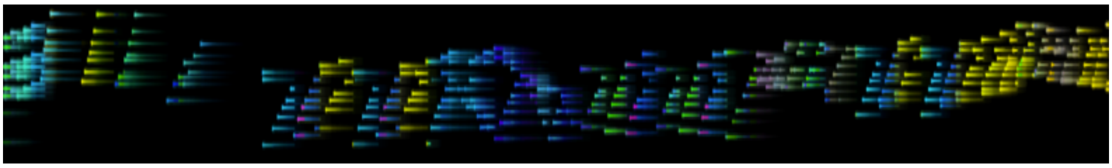


Figure 12 - Excerpt from Debussy's Clair de Lune (Ciuha, Klemenc, Solina, 2010)

2.1.2.9. Movie from Music: Visualizing musical compositions

In this project, they based on the association about colors and pitches. Therefore, they used a color raster graphics display to generate images based on their theory. The colors are blue, blue-violet, violet, red-violet, red, red-orange, orange, yellow-orange, yellow, yellow-green, green, and blue-green. Moreover, they used the raster display to create the "musical paintings". They use rectangular, square and circular format for the graphic implementation (Herman, Badler, 1979).

For this, their program requires the following information:

1. Number of chords or notes
2. Number of note for each chord
3. Notes in the chord
4. Number of lines in each color sector
5. Angle of separation between adjacent lines

With this knowledge, it is possible then to generate the visualizations according to it. The visualization must be coherent with the piece of music, additionally it has to be a pleasant presentation so people really enjoy it. The visualization should change according to the information received from the music, making it consistent with the music structure. Only when the visualization is coherent with the music, the audience will understand the feelings or message of the music that they are listening.

Here are some figures of the visualizations of this project:

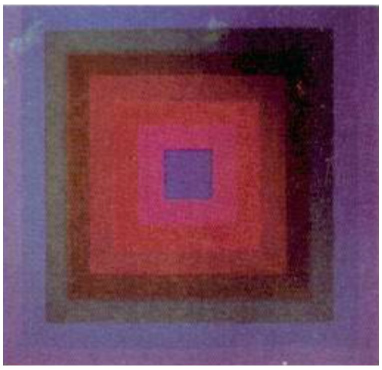


Figure 13 - Scale of blue violet (Herman, Badler, 1979)

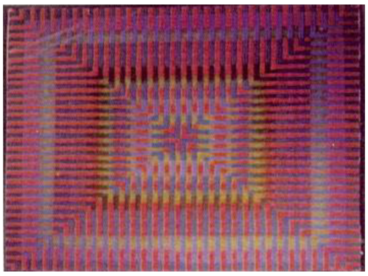


Figure 14 - Visualization for a two voices chant (Herman, Badler, 1979)

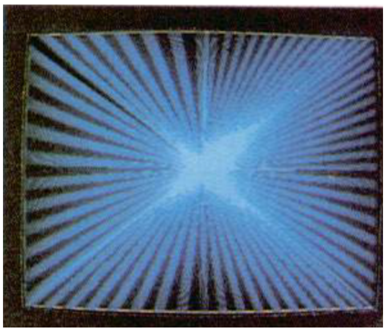


Figure 15 - Rectangular format (Herman, Badler, 1979)

2.1.2.10. Visualization of Music Performance as an Aid to Listener's Comprehension

In this project, Hiraga et. al focused on music elements such as tempo change, dynamics change, and articulation (rhythm and melodic structure) change. Those music elements are used to influence the visualization accordingly (Hiraga, Matsuda, 2004). Tempo is represented as a horizontal interval delimited by vertical lines. Dynamics changes are represented by the height of the bar, and articulation by the width of the bar. The gray scale of the graphical components represents the rhythm and melodic structure (see figure 16) (Hiraga, Matsuda, 2004).

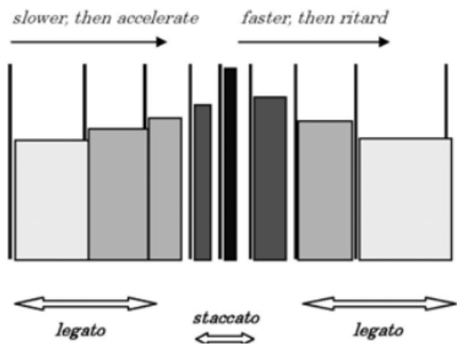


Figure 16 - Explanatory figure of music visualization (Hiraga, Matsuda, 2004)

As we can see in the figure above, when the tempo gets faster, the interval between the vertical lines along the horizontal axis is shorter, so when people see that, can associate the shortest intervals to a fast tempo (music rate). For the articulation, when is legato (the musical notes are played or sung smoothly and connected) the bars are close together. Giving the perception that, the sound is supposed to be connected. In the other hand, when the sound is suppose to be disconnected and short, we can observe that the bars are separated, this musical articulation is defined by staccato. Additionally, is possible to observe that this example represents a crescendo and the diminuendo, because dynamics are depicted by the height of the bars. In figure 16, the height of the bars is smaller at the beginning, and then is higher, this means that the volume is gradually higher (crescendo) and then the height of the bars is shorter again, in other words, the volume is lower (diminuendo).

2.1.2.11. Towards Building an Experiential Music Visualizer

In (Nanayakkara, Taylor, Wyse, Ong, 2007) is proposed a system that is capable of building real-time music visualizations rapidly using Max/MSP with Flash. They based on some music characteristics (pitch and note duration) to develop the music visualization scheme. In their system, they use a MIDI keyboard so they can have the knowledge of the music characteristics. The following figure (figure 17) shows the different audio-visual mapping for each instrument.

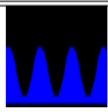
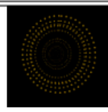
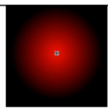
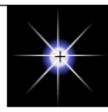
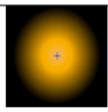
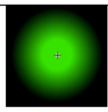
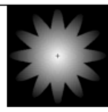
Instrument	Visual Effect	Instrument	Visual Effect
Bass Drum Effect: Wave (Pulsating up and down) Colour: Blue Screen position: Bottom edge of the screen		**Closed Hi-Hat** Effect: Bursting effect Colour: Yellow Screen position: Lower-Left of the screen	
Snare Drum Effect: Sphere fading away Colour: Red Screen position: Lower-Middle of the screen		**Ride Cymbal** Effect: Star falling down Colour: Silver Screen position: Top-Middle of the screen	
Hi Tom Effect: Sphere fading away Colour: Gold Screen position: Lower-Right of the screen		**Crash Cymbal** Effect: Firework effect Colour: Red Screen position: High-Middle of the screen	
Hi Bongo Effect: Sphere fading away Colour: Green Screen position: Lower-left of the screen		**Piano note** Effect: Star-like object fading away Colour and screen position: depends on the note class (C, C# ect)	

Figure 17 - Audio-visual mapping of different instruments (Nanayakkara, Taylor, Wyse, Ong, 2007)

The reason they used MIDI is because it is a lot easier to manipulate music information and it is a compact form data, which means that the MIDI files are smaller. The next figure (figure 18) illustrates the result of their work.

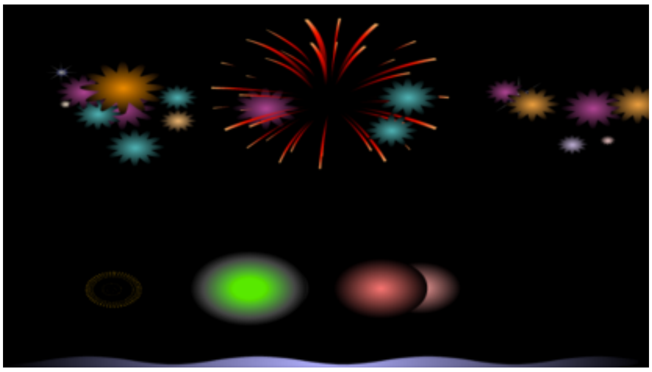


Figure 18 - Sample output of the experiential music visualizer (Nanayakkara, Taylor, Wyse, Ong, 2007)

2.1.2.12. A Visualization of Music

The goal of their project is to present an alternate method for visualizing music that makes use of color and 3D space. They defend that, most of the music listeners, are uncomfortable with music notation, in the other hand, music players, when look to the notes on a paper, they have the impression of how a particular piece of music will sound. Therefore, they wanted to create a system to help people who are not familiar with the music notation to better comprehend it through visual means (Smith, 1997).

They use MIDI because it contains the music information. Thus, the program parse input MIDI data files and generate graphical representations based on the music information. Their mapping function has two steps: mapping tone data, and mapping instrument data. The mapping tone data consists of representing the generated tones by the instruments, which are characterized by colored spheres. The spheres have special characteristics that can change according to their musical tone properties (pitch, volume, timbre). Firstly, the pitch is represented by the position along the vertical y-axis, secondly, the volume is represented by the size of each sphere and lastly, the timbre is represented by the color to show the distinctions between each instrument group (Smith, 1997).

An individual instrument plays a sequence of tone with specified rhythms throughout a musical piece. For the mapping instrument data, these tones produced by an instrument are represented by as a sequence of spheres of various shapes and sizes. The individual instruments are mapped to particular values along the horizontal x-axis.

2.2. Available Technologies

In order to search for available technologies in music visualization, it was decided to look for previous work about this thematic, and YouTube[1] website turn out to be a great source to watch videos about music visualization. People use this website in order to publish their works, and it was decided to explore about how those persons did those works, and what technologies they used. Researching background works about this theme is a plus, in order to know if any of those technologies are suitable for this thesis. So it was decided to make a list of those technologies and search what can be done, therefore it can be concluded if they are important for this thesis (why should they be used, or what for should they be used). The next list shows the technologies found:

Non-interactive technologies:

- Media Players supporting visualizations
- Adobe After Effects

Interactive technologies:

- VDMX
- Resolume Avenue
- Max/MSP
- Pure Data
- Quartz Composer

[1] www.youtube.com

2.2.1. Non-interactive Technologies

Media Players supporting visualizations:

There is some media player's software that supports music visualization. These media software plays audio files, and shows different visualizers for the current audio file that is being played. For instance, on iTunes, the user can enable the visualizer window on the tools bar, selecting the *view* option and then *show visualizer* (see figure 19).

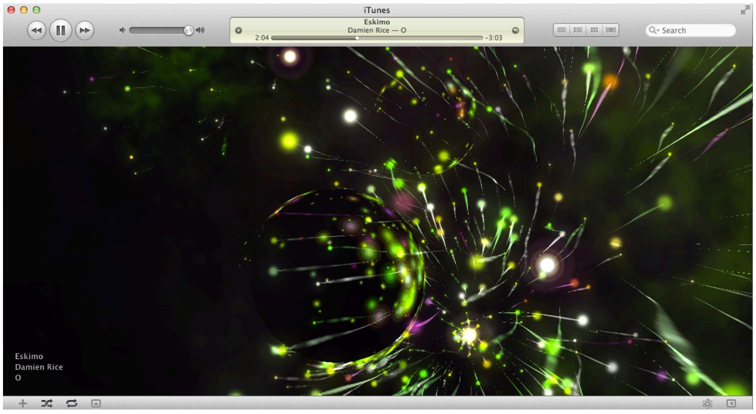


Figure 19 - iTunes visualizer

Other software that permits music visualization is "Windows Media Player" (Windows, 2012), "Winamp" (Nullsoft, 2012), and "VLC media player" (VideoLan Organization, 2012). All the visualizers have the same purpose, they react according to the music which is playing at the moment enhancing the listening experience of the listener. The visualizer generates animated imaginary, which is usually generated and rendered in real time and synchronized with the music file as it is played.

Adobe After Effects:

This software was designed for digital motion graphics, it is used in the post-production process of filmmaking and television production. With *Adobe After Effects* is possible to do 2D or 2.5D animations, visual effects (Adobe Systems Incorporated, 2012).The main interface consists of some panels (Project Panel, Timeline Panel, Composition Panel, etc.). The project panel is where is possible to import videos, audio footage items (media files). The timeline panel is where the time and the order of the layers can be adjusted. Lastly, the composition panel is where the items are visible at the current time maker (see figure 20) (Adobe Systems Incorporated, 2012).

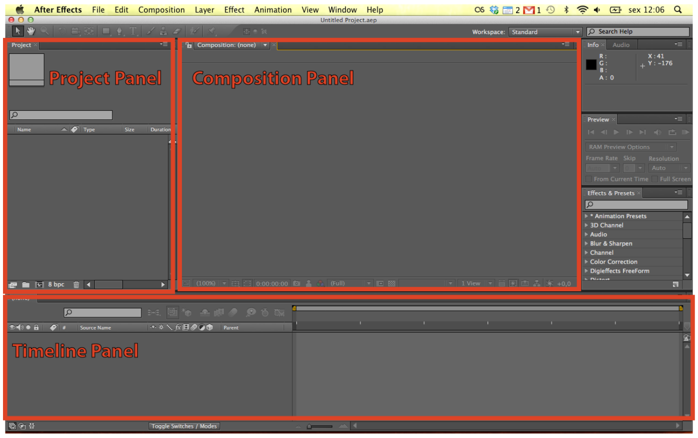


Figure 20 - Adobe After Effects Software with the main panels

2.2.2. Interactive Technologies

VDMX:

VDMX is Macintosh software that allows users to achieve good visual effects in real time (see figure 21). This software is appropriate for a musician who wants to add video to his sets, or a VJ working with a different DJ every night. With VDMX is easier to use audio and video together. Another advantage is the quality of the visual effects and real-time broadcast because this software uses the processing power of the computer's CPU. Finally, VDMX is compatible with Quartz Composer, which can be used for motion graphics visual development. In (Vimeo, 2012) we can see a video with one of the several examples of using a Quartz Composer patch in VDMX. This example is a bouncing ball that bounces with the music beat, with VDMX is possible to use the inputs of the Quartz Composer composition, and in this case, we are using the y position of the ball input to vary with the beat of the music that is being played. The disadvantage of this software is the price, it costs 300$! It's very expensive, so for this project it might not be used in consequence of the price (VIDVOX LLC, 2012).

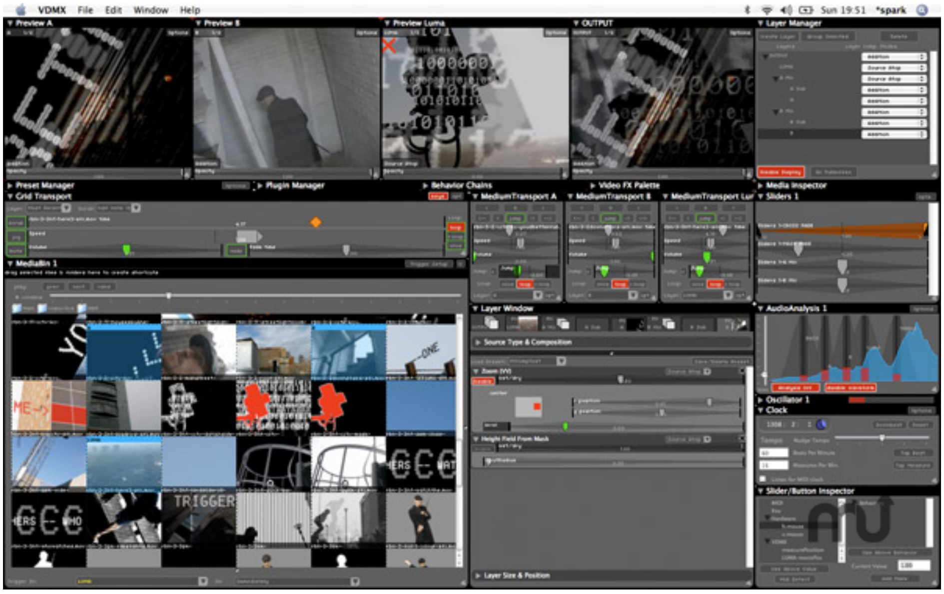


Figure 21 - VDMX software (DownloadAtoZ, 2011)

Resolume Avenue:

This software is very similar to VDMX, i.e., it is like an instrument for live performances with audio and also visual effects (see figure 22). It is a very powerful tool, which allows users (normally VJs) to do a very large number of stunning effects (Resolume, 2012). Therefore, it has the same advantages that VDMX has, and it is also compatible with Quartz Composer.

In (Wonder How To, Inc, 2011) we can see a video, that shows how we can integrate a Quartz Composer patch in Resolume, and we can see that it is quite simple. Avenue offers support for four different input parameters also referred to as published inputs in Quartz Composer (Number, String, Index and Boolean). The disadvantage of this software is the interface, because in my opinion it is a bit confusing for users who are not used to this kinds of technologies, and also it is expensive (but it has 50% discount for students). There is one version, which can be downloaded (version 3.3.2), but until the serial is not entered, it will show the Resolume logo and a robot voice in the audio.

Figure 22 - Resolume Avenue software (vBulletin Solutions Inc., 2011)

Max/MSP:

This is a visual programming language (such as Quartz Composer) for music and multimedia. This kind of software is mostly used by composers, performers, software designers, artists, etc. who are trying to create new, original and innovative performances (Cycling '74, 2012).

With Max is possible to create unique sounds or visual with “blocks”. These “blocks” are called objects, which are visible boxes that contain programs to do something specific, therefore, each object has a different program for different purposes (make noise, video effects) (see figure 23). The user just has to add these visual blocks to the canvas and connect them in order to do something that the user pretends (Cycling '74, 2012).

This software also allows connecting devices (live camera, audio input, Arduino, MIDI devices, USB gaming controllers) in order to help the user achieve better results, and expanding the functionality (Cycling '74, 2012). Most of the existing routines are shared libraries, as the software has 20 years, there are many routines available for everybody. Furthermore, there is the possibility of adding new extensions to Max/MSP, which can be written by third-party developers in C, C++, Java, or JavaScript (Cycling '74, 2012).

Nowadays, it is normal to use laptop computers in live music performances, which enhanced the Max/MSP development environment (Cycling '74, 2012). This kind of software is so evocative for artists in current days, because they can achieve another dimension of performance that attracts the audience and enhances the commitment level between the audience and the performance.

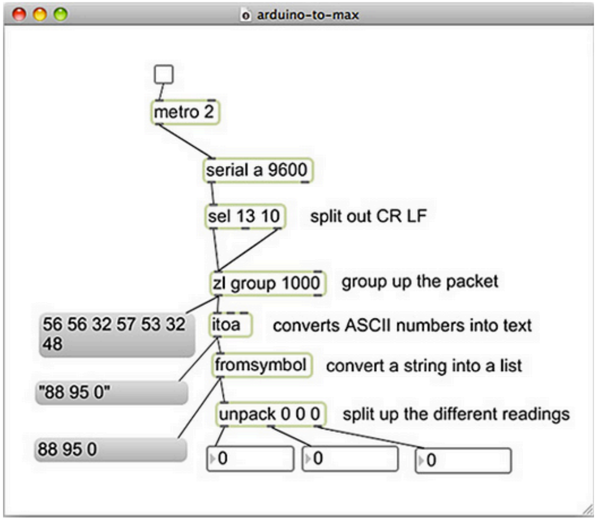


Figure 23 - Max/MSP software sample

Pure Data:

Pure Data is a real-time graphical programming environment (see figure 24) such as Max/MSP and Quartz Composer. Miller Puckette and company at IRCAM developed it, but nowadays it is maintained by Miller and includes the work of numerous developers (Pure Data, 2012).

This is a free software written for multi-platform therefore is somewhat portable. It is possible to write externals and patches that work with Max/MSP, and it is also possible to work with Quartz Composer through the OSC patch, which is responsible for opening a port and then send data through that port to Quartz Composer OSC receiver patch. Pure Data is natively designed to enable live collaboration across networks or the Internet, allowing musicians connected via LAN or even in distinct parts of the world to create music together in real time. The following figure (figure 24) illustrates an example with Pure Data software.

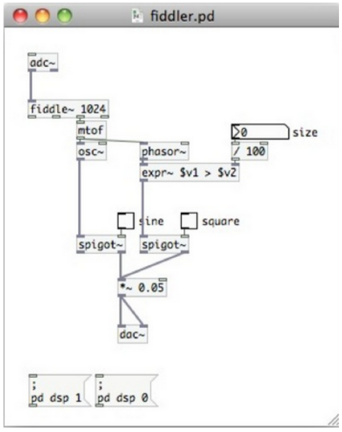


Figure 24 - PD software example

Quartz Composer:

Quartz Composer is a visual programming environment by Apple. This software allows the user to create motion graphics compositions without having necessary to write a line of code. That's the beauty of visual programming, the user only have to connect "blocks" that corresponds to graphics processing such as Pure Data. These "blocks" (patches) are functions that can affect images, sounds, etc. There are a lot of those functionalities, which makes the elaboration of dynamic visualizations with Quartz Composer flexible. It is also possible to build new blocks in *Xcode*, so the user can completely achieve his goals with motion graphics designing (Apple Inc., 2012). In the following section (section 2.3), it is explained in detail what is this software.

2.3. Quartz Composer

Quartz Composer is great software that enables the quick and easy creation of powerful animation graphics. With Quartz Composer visual programming environment is easier to create new kinds of visualizations for those who doesn't have much experience in a programming language. Furthermore there is a lot of documentation available on the Internet, due to the vast number of quartz composer users who shares experiences and knowledge about this software. In this section, it will be described in detail what is quartz composer, and what is possible to do and also how to do it.

2.3.1. Compositions

Quartz Composer compositions are the programs created in Quartz Composer software by adding patches (see section 2.3.2) in a workflow for data processing and rendering. The following figure (figure 25) shows a simple composition (Apple Inc., 2007).

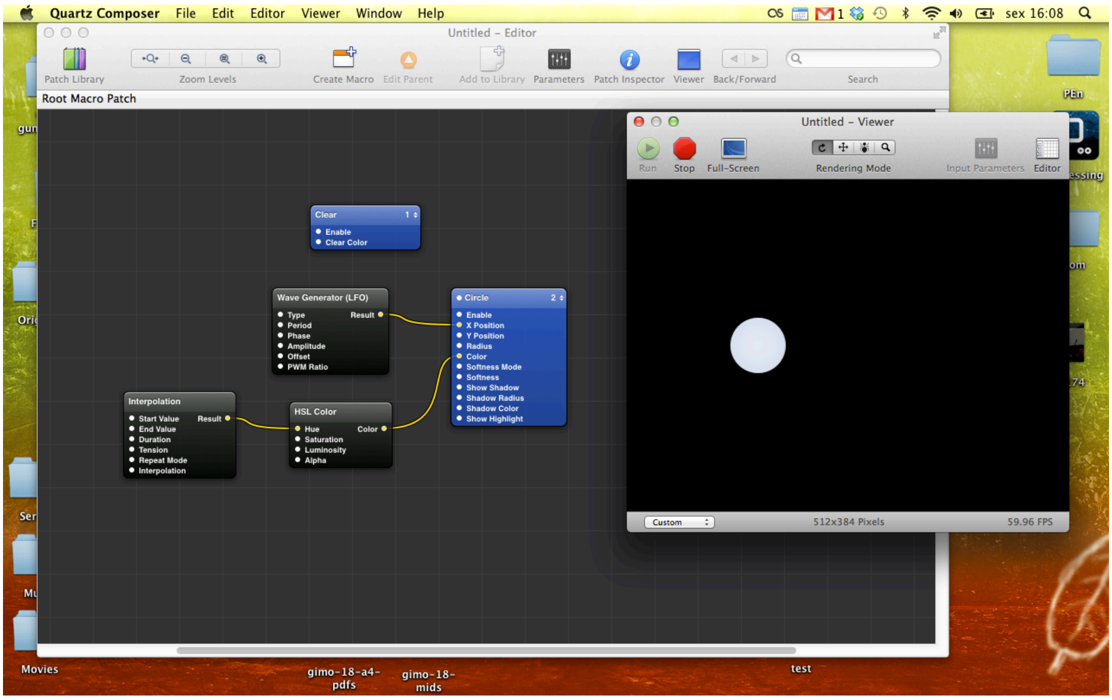


Figure 25 - Simple composition in Quartz Composer

In the previous figure (figure 25), we can observe a simple composition in Quartz Composer with only five patches. There is one patch to clear the layout, and then we have a circle patch. This circle patch renders a circle into the viewer window, and we can change its parameters. In this example, it was used other patches to change those parameters, specifically a "Wave Generator (LFO)" patch in order to change the x position of the circle, using the sin function of the patch, and then the "Interpolation" patch to create values from zero to one, connected to a "HSL Color" patch. Here, the "hue" parameter is changed according to the value of the "Interpolation" patch. Then, the resulting color will be the color of the circle, rendered in real time.

2.3.2. Patches

Patches are the basic elements of Quartz Composer. They correspond to regular routines in any traditional programming language, in other words they are processing units. This means that they execute during a certain time, and produce something (Apple Inc., 2007). As it is possible to see in figure 26, the circles are the ports of the patch, which corresponds to parameters in functions of a programming language. It is also possible to notice on the figure that not every patches has input (on the left side) and output (on the right side) ports. As it is visible on figure 26, we have the "Audio Input" patch that has both output and input ports. The input ports are the "Increasing Scale" port and the "Decreasing Scale", these ports have as objective to define the scaling factor peak and spectrum bands variation when positive or negative, respectively. The output ports are "Volume Peak" and "Spectrum". The "Volume Peak" corresponds to the maximum value of the audio data analyzed at the moment, and the "Spectrum" corresponds to the audio spectrum. The "Clear" patch only has two input ports, which are "enable" and "Clear color". The enable parameter is used to control execution of the patch, and the "clear color" corresponds to the clearing color of the layout. Finally, we have the "Kineme Audio Device Info" which provides information about the available hardware audio devices. This patch has only two output ports (Input Devices and Output Devices), with the purpose of giving information about the input audio devices or the output audio devices.

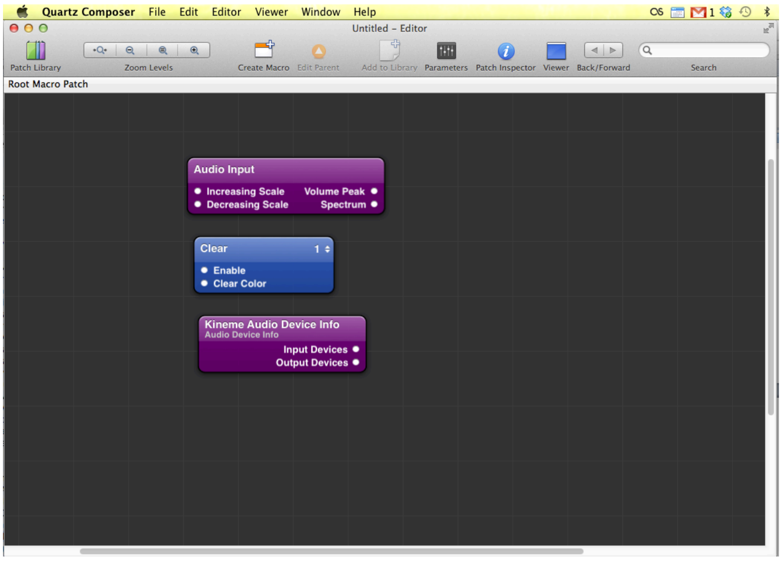


Figure 26 - Quartz Composer Patches

In order to add new patches to the Quartz Composer composition, there is the "Patch Creator" (see figure 27), where is possible to browse the existing patches, and with a click on the name of the patch, it is possible to see the information (a resume) of what that patch does (Apple Inc., 2007).

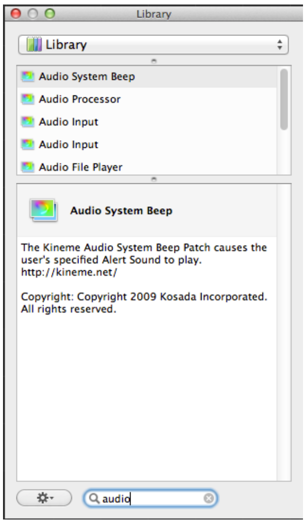


Figure 27 - Patch Creator

There are several types of patches in Quartz Composer: Composite Patches, Controller Patches, Environment Patches, Filter Patches, Generator Patches, Gradient Patches, Modifier Patches, Network Patches, Numeric Patches, Plug-in Patches, Programming Patches, Render Patches, Source Patches, Tool Patches, and Transition Patches (Apple Inc., 2007).

The **Composite Patches** are those that have two input image ports and one output image port (see figure 28), which are used for compositing and blending operations (addition, source over, source out, etc.) (Apple Inc., 2007).

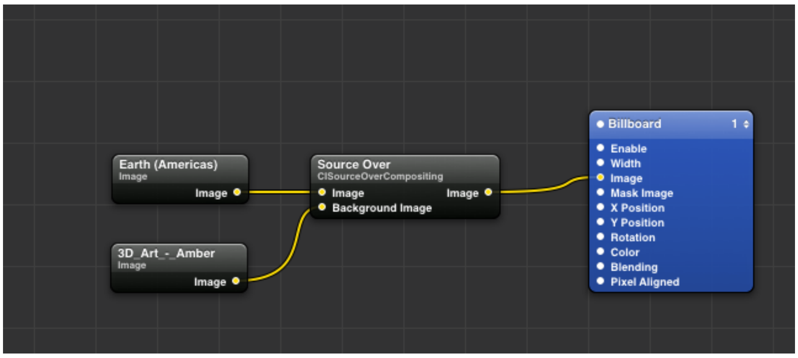


Figure 28 - Sample with a composite patch (Source Over)

In the previous figure, the “Source Over” composite patch is used to put one image over the other.

The **Controller Patches** produce one or more values that could be from the output of another patches. These patches produce output values based on their input values (Apple Inc., 2007). The figure 29 illustrates a controller patch (Mouse Patch), which is used in this case to control the position of the cube through the coordinates (x and y) of the mouse.

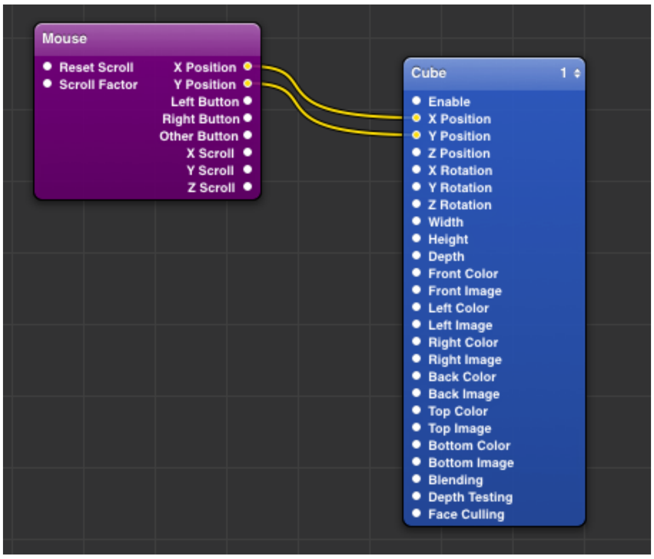


Figure 29 - Mouse Patch is a Controller Patch

Environment Patches are normally macros that operate on motion graphic to transform its appearance, like for example the "Lightning Patch" or the "Fog Patch" (see figure 30) (Apple Inc., 2007).

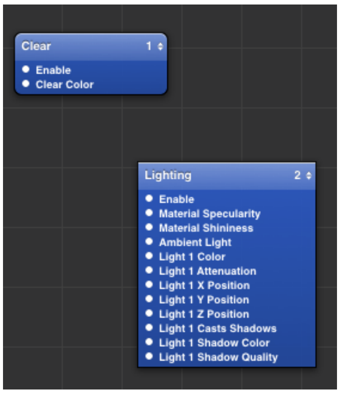


Figure 30 - Environment Patch

The **Filter Patches** operate on the pixels of the input image and produce a different output image (Apple Inc., 2007), for instance, in figure 31 the "Pixellate" patch will make the "foto_cv" image blocky.

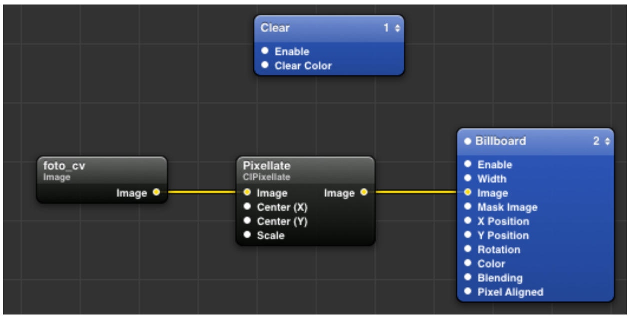


Figure 31 - Filter Patch (Pixellate)

The **Generator Patches** (figure 32) produce an image algorithmically, for example, in the following figure, the "Checkerboard" patch creates checkerboard pattern whose colors, size, square size and sharpness depends on the input values (Apple Inc., 2007).

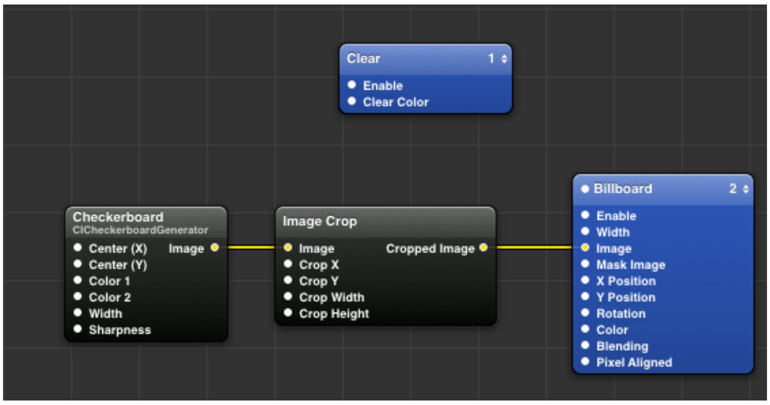


Figure 32 - Generator Patch (Checkerboard)

The **Gradient Patches** (see figure 33) produces an image based on Gaussian, linear, or radial algorithms and the output image can be used as input in another patch (Apple Inc., 2007).

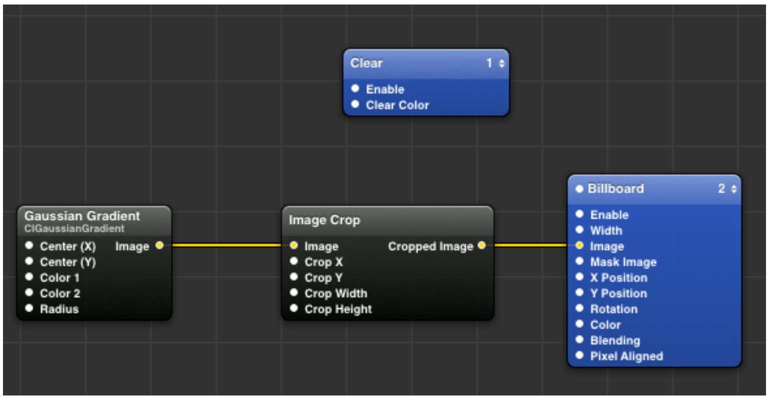


Figure 33 - Gradient Patch (Gaussian Gradient)

The **Numeric Patches** produces numerical values based on mathematical expressions (Apple Inc., 2007). In the following figure (figure 34), there is a mathematical expression (a*(1 + sin (b))), where "a" and "b" are the input values of the patch, and can be changed, or added more inputs.

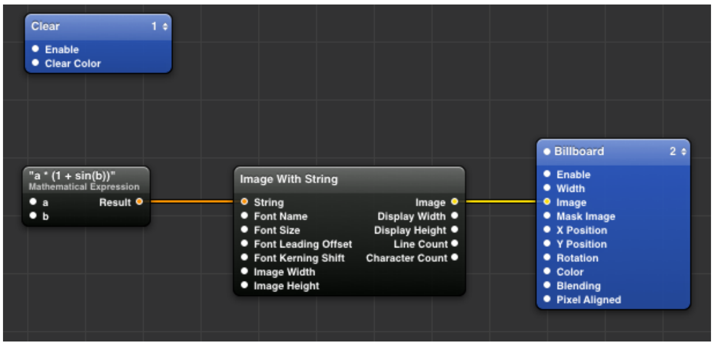


Figure 34 - Numeric Patch (Mathematical Expression)

The **Plug-in Patches** are those patches package as Quartz Composer plug-in, that were built by third parties (see section 2.3.7 to see how to do) (Apple Inc., 2007).

Figure 35 - Plugin Patch (_1024_GL_Circle)

In the previous figure (figure 35), we can see the "_1024_GL_Circle" patch, which is a patch available on (Kosada Incorporated, 2011) which renders a circle on the viewer window according to the input ports values.

The **Programming Patches** provide a text field so the user can write a new routine that he thinks is necessary for his composition, for example the "JavaScript Patch", which can be used to write a JavaScript routine (Apple Inc., 2007).

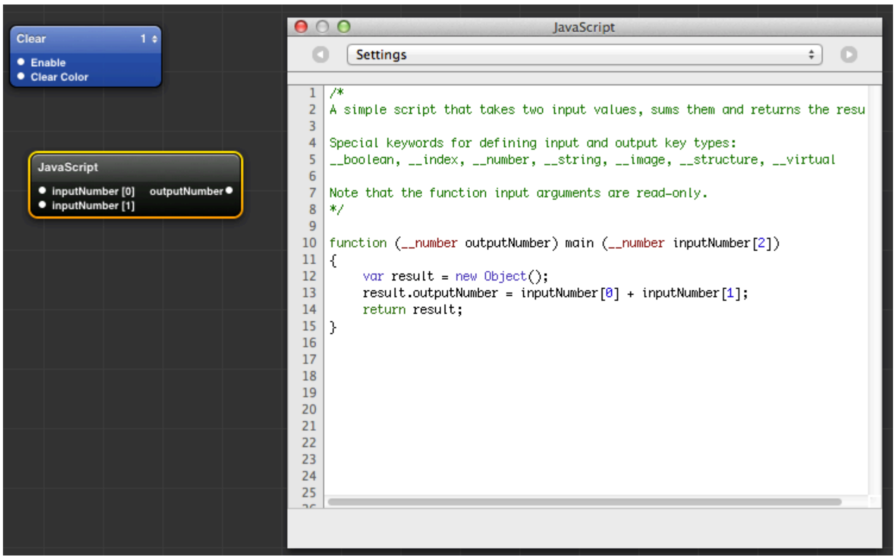


Figure 36 - JavaScript Patch (Programming Patch)

"JavaScript" patch is one of the available patches on Quartz Composer, in the above figure (figure 36) we can see the window where we can really write code. The user just has to click on the patch and the press "cmd + 2" in order to program with the "JavaScript" patch.

The **Render Patches** render to the viewer window something, for instance, the "Billboard" patch renders a 2D image, the "Cube" renders a 3D cube (see figure 37), etc. (Apple Inc., 2007)

Figure 37 - Render Patch (Cube)

The **Source Patches** provides data from a source like a file list, or the information of a composition source (see figure 38) (Apple Inc., 2007).

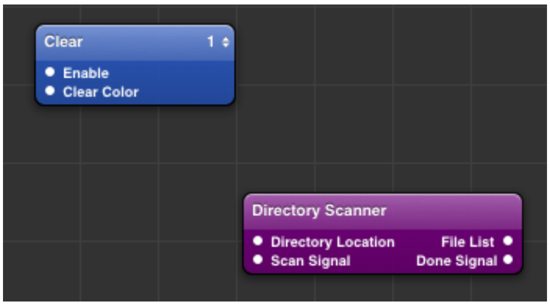


Figure 38 - Source Patch (Directory Scanner)

The following figure (figure 39) shows a different kind of patch: macro patch. These macro patches are patches that can include others inside it, in other words, it is like subroutines in a traditional program, additionally it is possible to call another macro patches, this means that macro patches can be nested, forming a **patch hierarchy** (see figure 43) (Apple Inc., 2007).

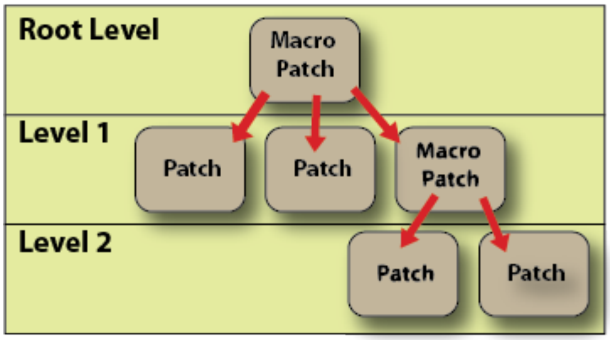


Figure 39 - Quartz Composer Patch Hierarchy

In the next figure (figure 40), it is possible to observe the evaluation order of the patches. There are kinds of patches (consumer patches) that are evaluated by a special order. Those patches have a number, which determines their **evaluation order**, which is from the lowest numerical number to the highest (Apple Inc., 2007).

As you can see in the figure 40, there are two consumer patches (clear and cube), and those patches are following a certain order depending on their number, so in this case, the first patch that is going to be the first one to execute is the "clear" patch. Then, the second will be the "cube" patch. In order to "cube" patch complete its execution, the patch receives data from the "Audio Input" patch in order to determine the Y position (the Y Position will change with the peak of the audio), and then the X rotation of the cube will vary according to the output value of the "Wave Generator (LFO)" patch. After the "cube" patch have all data from the other patches, it can finally render its result.

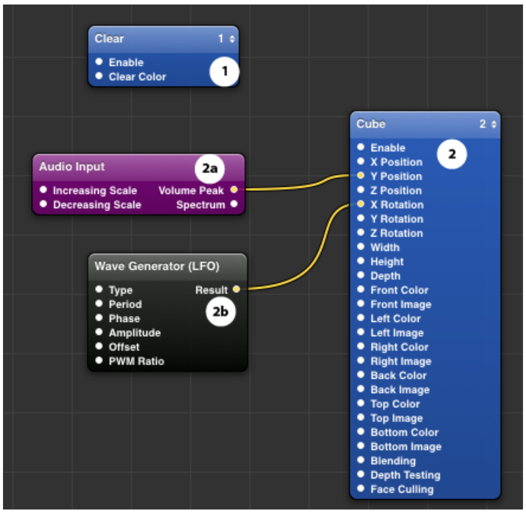


Figure 40 - Quartz Composer Evaluation Order

2.3.3. Programing Patches

There are three types of programming patches in Quartz Composer: Core Image, JavaScript, and OpenGL making this visual programming language more flexible.

The **Core Image Filter Patch** is very useful for image processing filters. To use this patch, the user has to be familiar with Core Image Kernel language, which is an extension to the OpenGL Shading Language (Apple Inc., 2007).

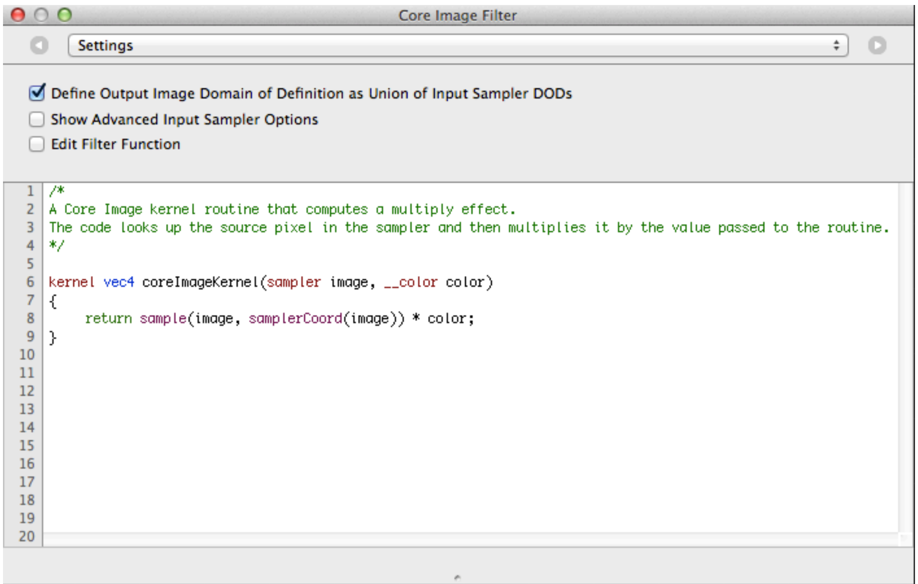


Figure 41 - Core Image Filter Patch

In the previous figure (figure 41), we can see a simple example with the "Core Image Filter" patch. In this example, there are two parameters, which are an image, and a color, and then this routine will multiply the source pixel by the value of the color.

The **GLSL Patch** uses the vertex and fragment shaders the user provides to render a result (Apple Inc., 2007).

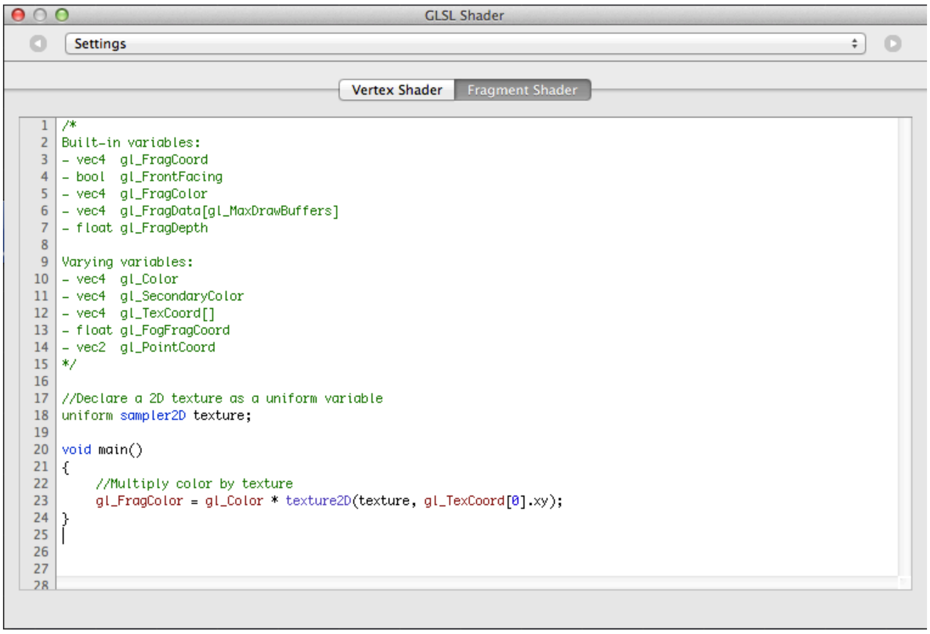


Figure 42 - GLSL Shader Patch

In figure 42, we can see an example of a GLSL Shader routine, which multiplies the color by the texture. The **JavaScript** runs the function written by the user in the Settings Pane of the patch. It is automatically created input and output ports for the patch, also it is necessary to define the ports with: __number (double precision floating point number), __index (positive integer between 0 and 2147483647), __string (unicode string), __image (2D image of arbitrary dimensions), __structure (named, or ordered collection of objects, including nested structures), and __boolean (boolean value, 0 or 1) (Apple Inc., 2007).

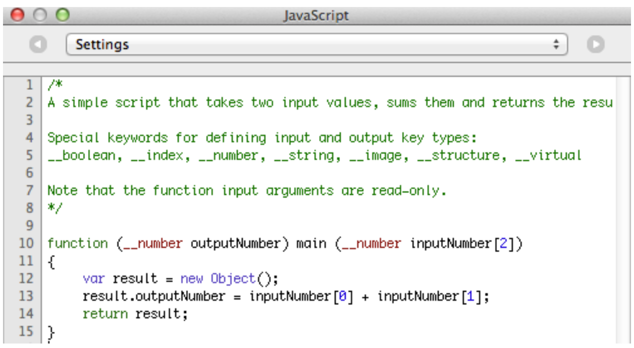


Figure 43 - JavaScript Patch

In the above figure (figure 43), this simple JavaScript routine serves to add the two input numbers.

2.3.4. Coordinate System

Quartz Composer uses a three-dimensional coordinate system (see figure 44), where the origin is at the center of the screen, the x-axis is horizontal, and the y-axis is the vertical. The left and right borders of the x-axis have the coordinates -1 and 1 respectively. The coordinates of the maximum and minimum y-axis are 1/AR (Aspect Ratio) and -1/AR respectively. For example, if the aspect ratio is 4:3 than the maximum y-axis value is 0.75 and the minimum it will be -0.75. The z-axis is perpendicular to the screen (Apple Inc., 2007).

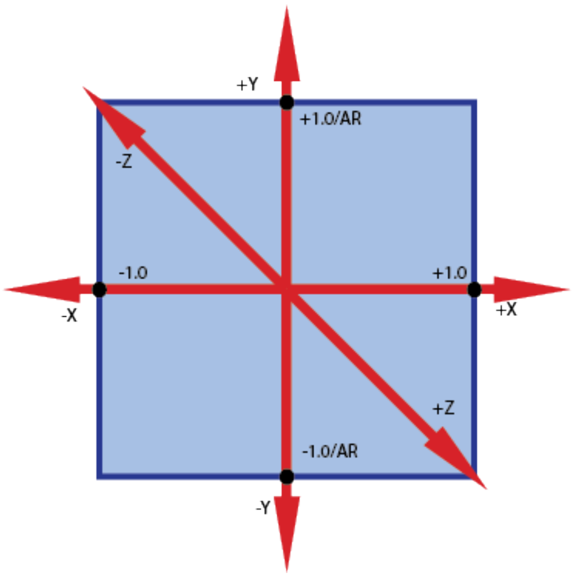


Figure 44 - Quartz Composer Coordinate System

2.3.5. Available Templates

Quartz Composer is a tool that helps with anything related to motion graphics, such as graphic animations, or graphic transitions, or music visualizers, etc. Therefore, there are several templates in Quartz Composer (see figure 45), which helps the user with his compositions about these categories described above (graphic transitions, music visualizer, etc.). These templates creates Quartz Composer files already with some patches that will help in a specific composition that the user wants to do, such as: Basic Composition, Graphic Animation, Graphic Transition, Image Filter, Music Visualizer, RSS Visualizer, Screen Saver, and Mesh Filter (Apple Inc., 2007).

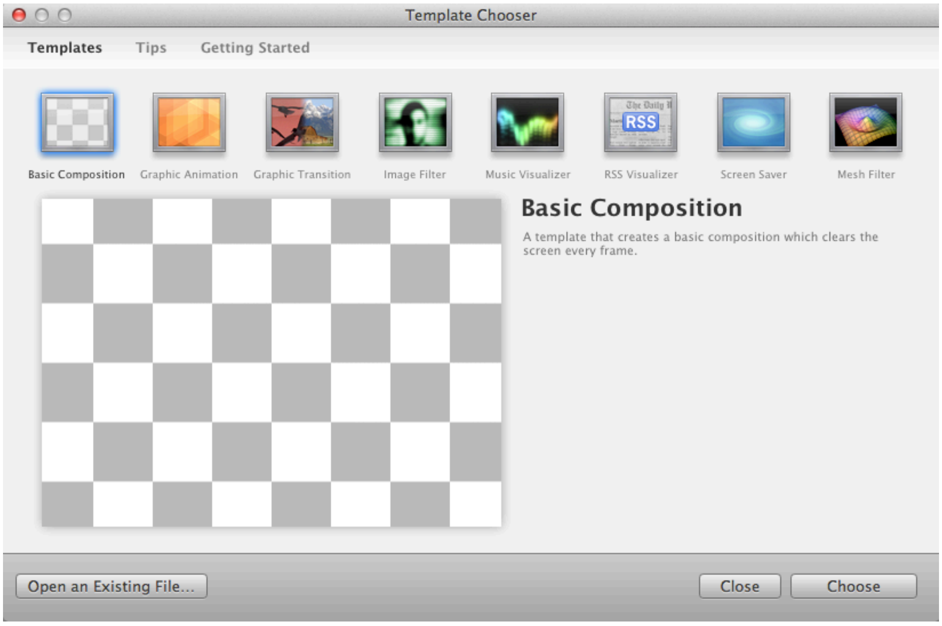


Figure 45 - Available Quartz Composer Templates

The **Basic Composition** template creates a basic composition, which clears the screen, every frame (see figure 46).

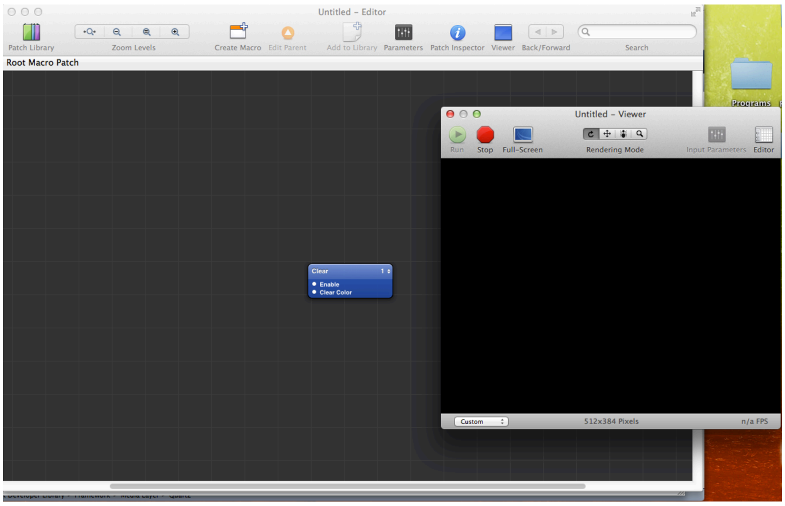


Figure 46 - Basic Composition Template

The **Graphic Animation** creates a file with the goal to render something to the screen, it is used to create generic motion graphic compositions. An example of this template could be the background animation (see figure 47).

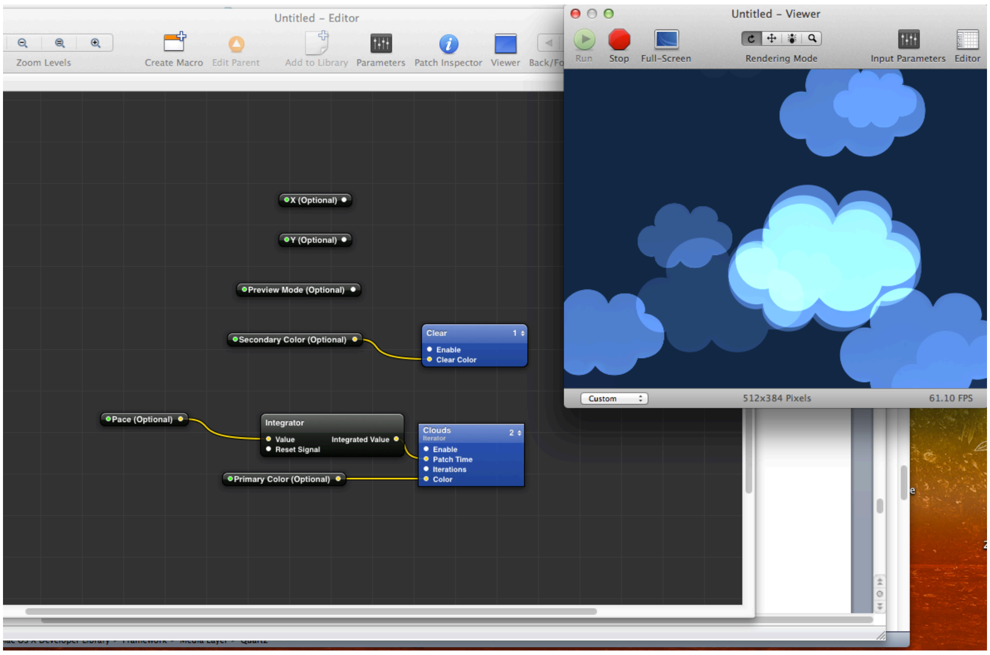


Figure 47 - Graphic Animation Template

Graphic Transition (see figure 48) is used for a composition that requires a transition between the source images to a destination image. So, for the template it is required a source image (which corresponds to the first image that will be shown) and destination image (which will the resulting image).

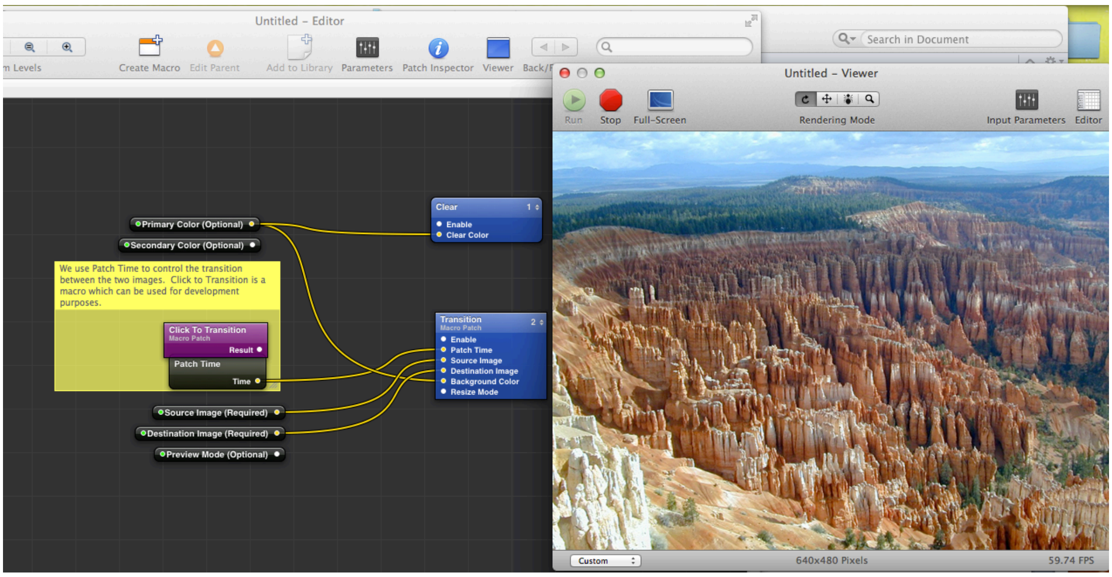


Figure 48 - Graphic Transition Template

Image Filter is used for compositions that are necessary visual effects on images. Therefore, is required an input image to apply the visual effect (see figure 49).

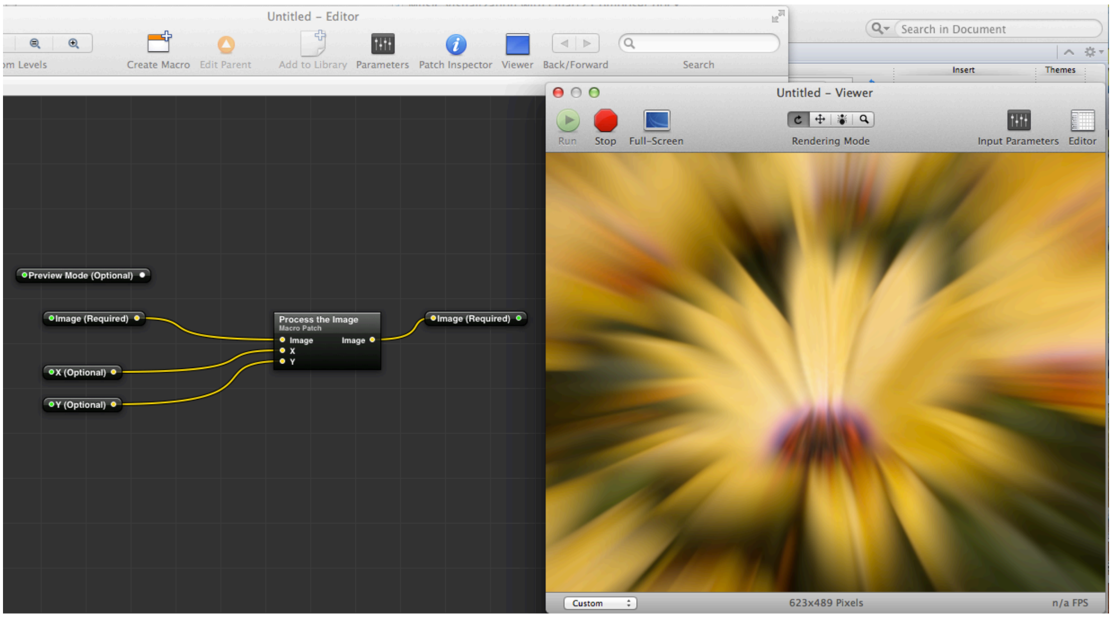


Figure 49 - Image Filter Template

The **Music Visualizer** is used for those compositions with visual effects affected with the music. These compositions are used for music visualizers on iTunes, or in this project, for a live performance (see figure 50).

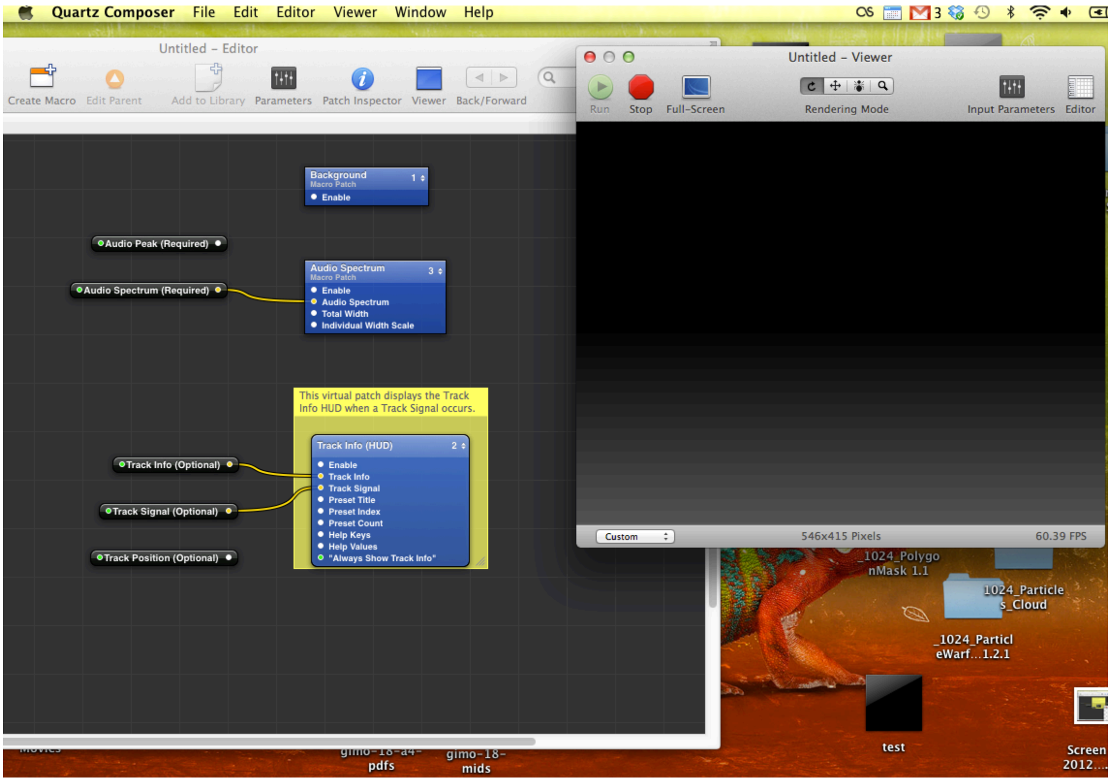


Figure 50 - Music Visualizer Template

2.3.6. User Interface

Quartz Composer has two main windows, which are the editor window and the viewer window. The editor window is the composition workplace, where the user adds the patches and connects them in order to do something (see figure 51). The viewer window (see figure 52) is where the user observes the result of the their composition. It is possible to see the rendering resources consumed by the composition, and the debugging information (Apple Inc., 2007).

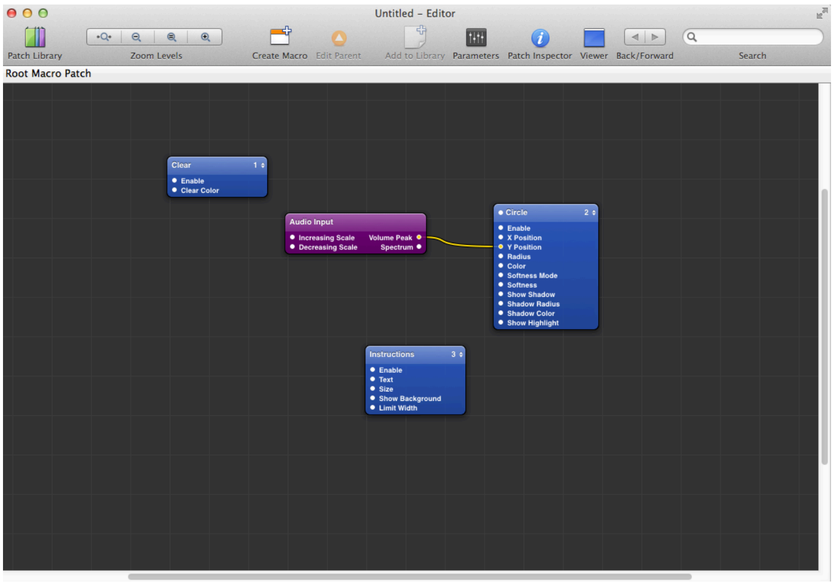


Figure 51 - Quartz Composer Editor Window

Figure 52 - Quartz Composer Viewer Window

In Quartz Composer, there is also a patch parameter pane, which enables the user to modify the attributes of a specific patch. Therefore, the user have to select the patch, and then in the patch parameter pane (see figure 53) he can see the values of the input parameters, and if he wants, he can change them, or he can also use the patch inspector for this purpose (see figure 54). The patch inspector also have other options (see figure 55), those options are input parameters, settings, and published inputs & outputs. Depending on the patch, the user might want to use these options, for instance, in this case, the user will use the input parameters to change the parameters of the "circle" patch. However, if the user was using a "JavaScript" patch (or other programming patch) he needs to use the settings option to program the pretended routine. The settings option corresponds to the definitions of the selected patch. Lastly, we have the published inputs & outputs option, where the user can see all the inputs or outputs that he published in that patch (users publish an input or output parameter in order to use the values of nested patches within a composition).

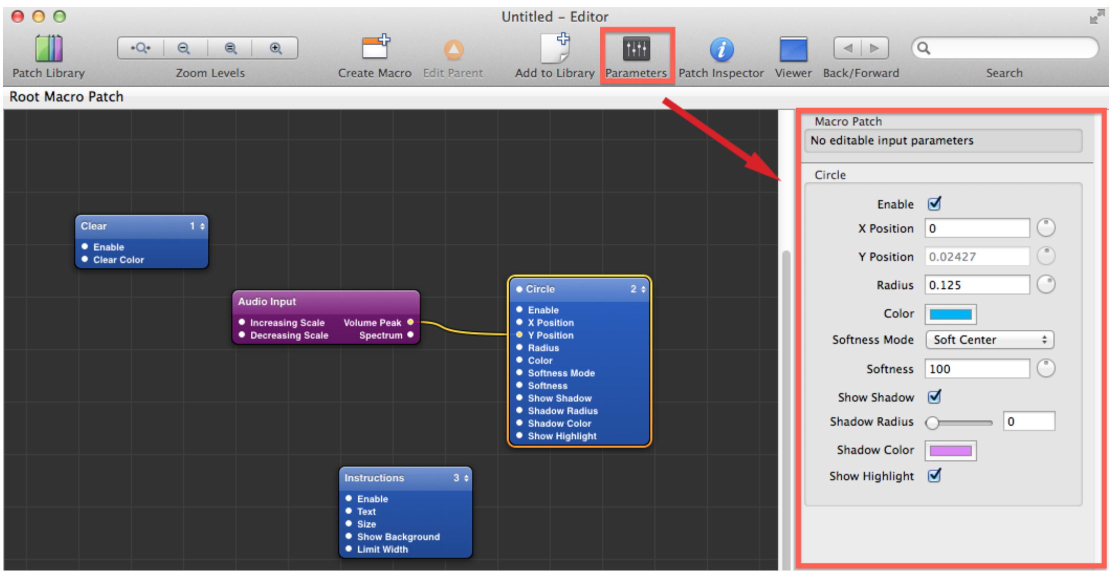


Figure 53 - Patch Parameter Pane

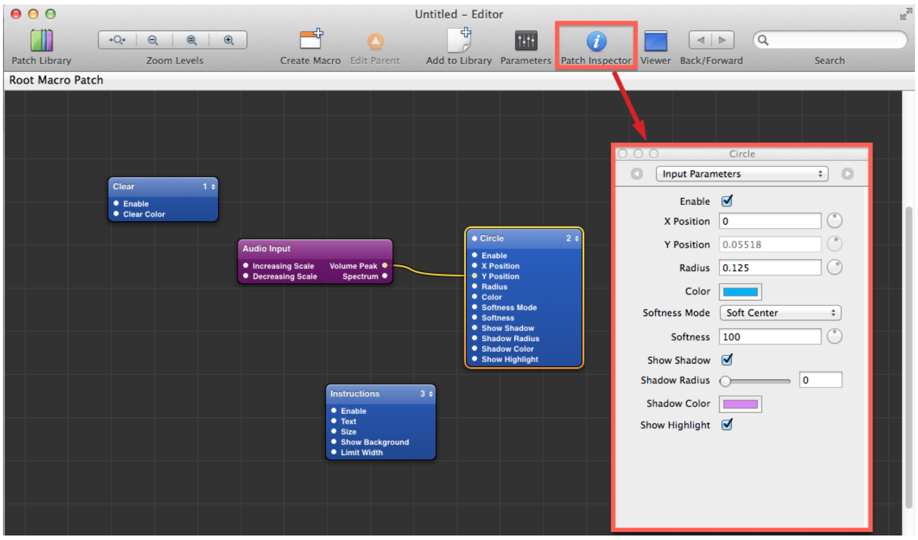


Figure 54 - Patch Inspector

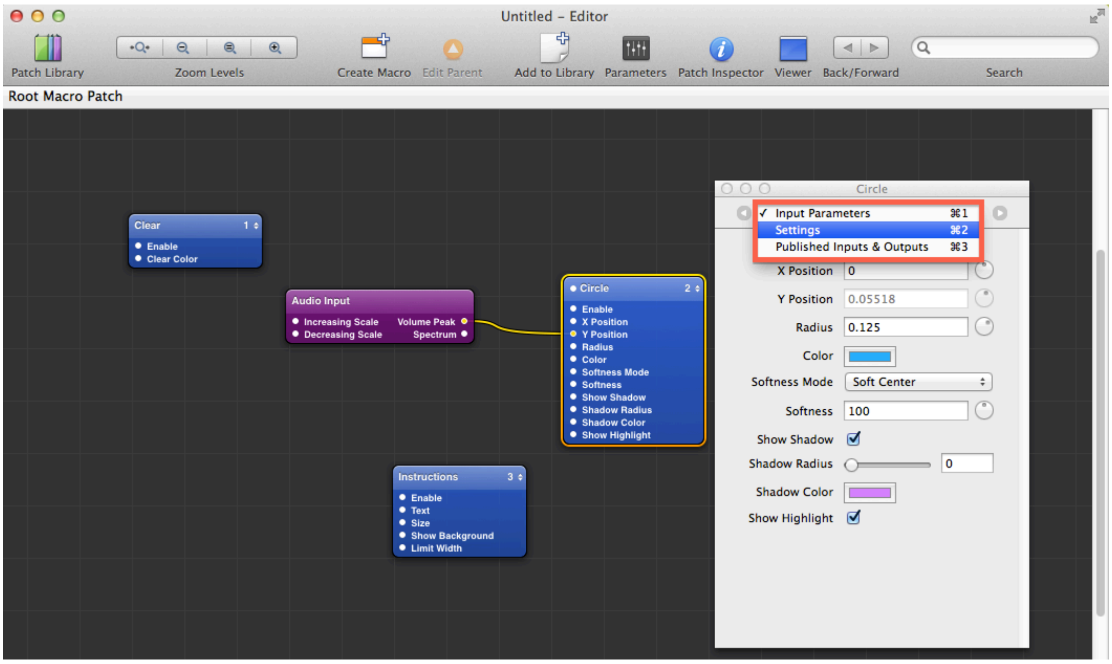


Figure 55 - Available options on the patch inspector (red rectangle)

2.3.7. Building Custom Patches

As it was said before, Quartz Composer is a flexible tool, because permits the user to create new patches, so he can achieve his goals. In order to create new custom patches, it is necessary to use Xcode tool and to write the code in objective-C programming language. Then the user creates a new Quartz Composer Plug-in project and then he can work on the Xcode template for the plug-in. In the template, exist an .h file and an .m file. The .h file automatically subclasses QCPlugIn and declares two string properties for the input and output parameters (Apple Inc., 2008), as you can see in figure 56.

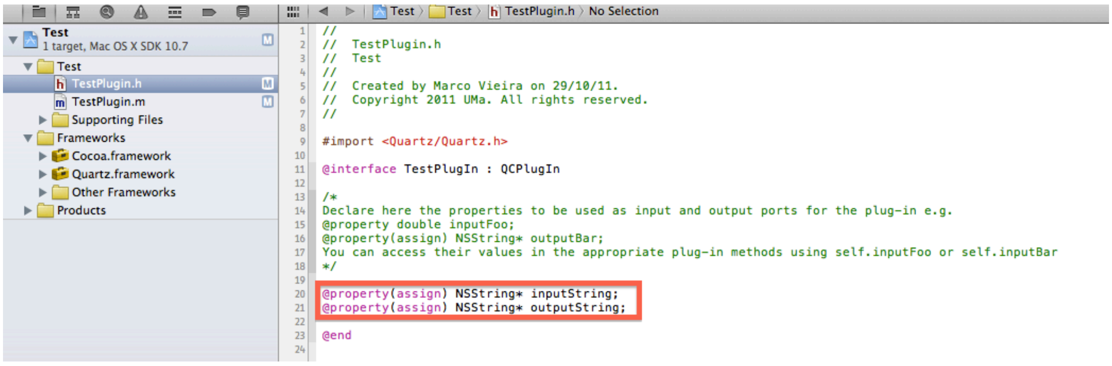


Figure 56 - .h file for Quartz Composer plugin

The .m file is where all the behavior is implemented, and some methods are automatically created. To build de plug-in, the user just has to modify those methods in the .m file. Those methods are: attributes, attributesForPropertyPortWithKey, executionMode, timeMode, and execute (Apple Inc., 2008). The **attributes** method (see figure 57) returns a Dictionary with all the attributes that can be accessed by their key (Apple Inc., 2008).

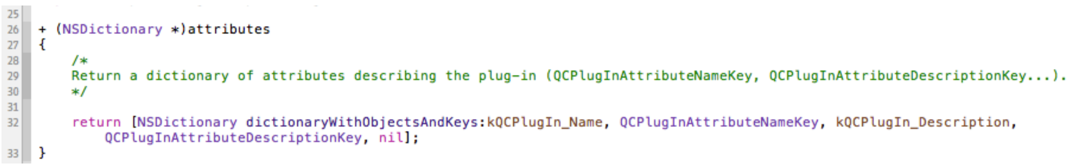


Figure 57 - attributes method

The **attributesForPropertyPortWithKey** method (see figure 58) returns a dictionary for each input and output parameter. Furthermore, each dictionary contains a value with its port attribute name key (Apple Inc., 2008).

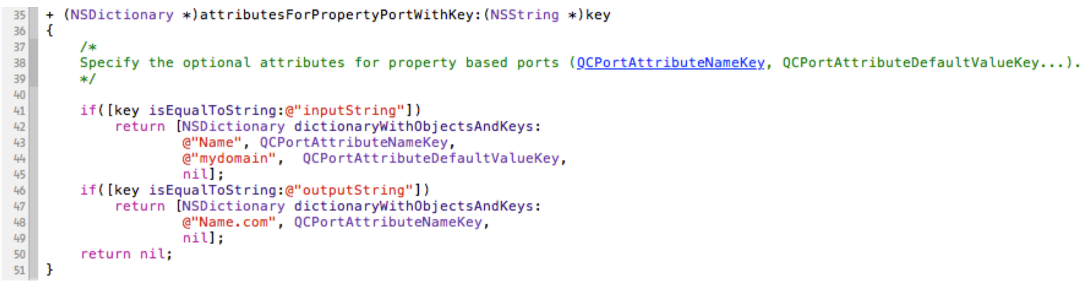


Figure 58 - attributesForPropertyPortWithKey method

The **executionMode** method (see figure 59) takes the input values, process them, and then outputs the result (Apple Inc., 2008).

```
+ (QCPlugInExecutionMode)executionMode
{
    /*
    Return the execution mode of the plug-in: kQCPlugInExecutionModeProvider, kQCPlugInExecutionModeProcessor, or
        kQCPlugInExecutionModeConsumer.
    */

    return kQCPlugInExecutionModeProcessor;
}
```

Figure 59 - executionMode method

The **timeMode** method (see figure 60) is the one that returns the time mode of the patch, and it's used to define if the patch depends on time or not (Apple Inc., 2008).

```
+ (QCPlugInTimeMode)timeMode
{
    /*
    Return the time dependency mode of the plug-in: kQCPlugInTimeModeNone, kQCPlugInTimeModeIdle or kQCPlugInTimeModeTimeBase.
    */

    return kQCPlugInTimeModeNone;
}
```

Figure 60 - timeMode method

The **execute** method (see figure 61) is where all the processing of the patch will occur. Here is where the logic of the patch has to be implemented (Apple Inc., 2008).

```
- (BOOL)execute:(id <QCPlugInContext>)context atTime:(NSTimeInterval)time withArguments:(NSDictionary *)arguments
{
    /*
    Called by Quartz Composer whenever the plug-in instance needs to execute.
    Only read from the plug-in inputs and produce a result (by writing to the plug-in outputs or rendering to the destination OpenGL
        context) within that method and nowhere else.
    Return NO in case of failure during the execution (this will prevent rendering of the current frame to complete).

    The OpenGL context for rendering can be accessed and defined for CGL macros using:
    CGLContextObj cgl_ctx = [context CGLContextObj];
    */

    self.outputString = [self.inputString stringByAppendingString:@".com"];

    return YES;
}
```

Figure 61 - execute method

2.4. General Discussion

To summarize, merging music could enhance music listening experience and can help the audience understand the existent feelings of music through visuals means. As discussed before, there are some different interesting approaches for music visualizations. However, they are not designed for live music performances (e.g. Lillie project entitled "Visualizing music"). Not all projects of music visualization are about creating visualizations for live music. The ANIMUS project (Taylor, Boulanger, Torres, 2005) is simply a fictional character that shows the feelings of a music piece through its expressions. In this project they have to detect happy and sad feelings within the music, and then translate it to a happy emotional response by the ANIMUS or a tragic emotional response by the ANIMUS in case if is detected a sad feeling. These happy feelings or negative feelings in music are detected by tonality. And this is an interesting fact, because in other projects "Emotion-base music visualization using photos" by Chen, et al, or "Interactive Music Visualization by Kulbelka, or "visual and aural – visualization of harmony in music with color" by Klemenc et al, and others projects they all do the same. So it was decided for this project, to base in western music and detect the feelings of music through the tonality, this means, that if it is detected a major chord, then it is a cheerful feeling, otherwise it is an unhappy feeling.

Lillie's project (Lillie, 2008) has another different purpose, which is educating people novice in music. With that project, it is possible to understand the duration of the different musical notes, all different pitches, etc. A person without any experience in the music world is incapable of reading a musical score. But, instead of trying to understand the musical score, they have a visualization, which illustrates the music of a musical score in a much easier way to understand. When they look for a musical score, people will not understand what is the duration of each musical note, such as other music elements. In Lillie's project, anybody understands the duration a musical note, and also other musical elements, which is great for novice people, giving them an opportunity to learn music in a much easier way.

Besides Lillie's project, there is Hiraga's et al project, which also has an educational goal. But they have a different approach, their visualization is represented by graphics, where the height of the columns of the graphics depicts dynamics (as it was explained previously), and other music elements are also represented by graphics properties (Hiraga, Matsuda, 2004). Nevertheless, Lillie's project is more intuitive because its design helps people to have a better understanding of the elements of music (volume, duration), rather than graphics, which is more difficult to interpret by audience in general.

Anyway, the work documented in this thesis has a different goal, which can be achieved by creating a system capable of responding to music in real-time environment. The graphic animations should be able to enhance the motive and dynamic qualities of a live music performance, capturing music elements while the audience listens the music and triggers several visuals related to those elements. The reported related works are not used in real time environments, and they use MIDI inputs (e.g. (Nanayakkara, Taylor, Wyse, Ong, 2007)) instead of non-digital sound. This project is designed for an orchestra live concert, therefore MIDI is not used, only non-digital instruments (e.g. mandolin, violin).

In order to achieve the goal of discovering the best solution of how should music be mapped into visuals, it was considered the mapping used in earlier works. With this research, it is possible to know if the existing music visualizations really convey the message or feelings inherent to music. In (Hiraga, Mizaki, Fujishiro, 2002), (Herman, Badler, 1979), they relate pitch with color, this is why at the beginning of this thesis this was one of the mapping functions used in the visuals. And there is also another method, in (Kubelka, 2000), (Klemenc, Ciuha, Subelj, 2011) and (Ciuha, Klemenc, Solina, 2010), they all relate color to tonality rather than pitch. Those papers, related tone to color, because as it was mentioned before, tone represent the mood of the music, and the color should be the visual element to represent the atmosphere of the music. Later on, the mapping between the music mood and color was created, because a single pitch don't transmit the music feeling, it is necessary to use the tonality and relate it with color.

In (Smith, 1997), they write about the mapping of other music element: volume - size. Volume is one the music elements most present in music and that anybody can understand it. So this is an important music element to be presented in the visuals, and it should be intuitive. The reason why volume is related to size is because when people listen a high volume, they associate it to something big, in the other hand, they associate the low volume to something small. Consequently, the elements of the visualization will look bigger when the volume increases, else they will look smaller.

In some projects, they use geometrical figures in order to give a better perception of all these music elements, for instance, in (Smith, 1997) it is used sphere and its parameters (position, size, color) are controlled by the detected music elements. In this case, they map the pitches into position in the y-axis, the volume into size, and timbre into color. In this thesis, this hypothesis of using geometrical figures is also going to be considered and tested in a live music performance with music visualization.

In Smith's project, it is also mapped the timbre to color, instead of pitches or tonalities. The timbre consists of the different sounds of each different instrument, so in these visuals the audience can understand all different instruments by the different colors present in the visualization. In (Nanayakkara, Taylor, Wyse, Ong, 2007), they also capture the different instruments and depict them with different visual effects for each different instrument. This is another interesting methodology, yet those visual effects should react to music as a whole, rather than individual instrument. Imagine, if a certain instrument has a small part in the music, then the visual of that instrument will not be presented, and the visualization can then become poorer. If the visualization is reacting to the music, there is always something happening, could be just one instrument, or several. But, the graphic animation will be richer by this way.

In a nutshell, the computer should make decisions autonomously during the creation and execution of the animations. In order to make this possible, there are some technical challenges such as real-time audio analysis. This system needs to provide an accurate audio analysis, manipulate audio data according to the creator's definitions and to generate in real time, which will be explained in section 4.2.

3. Music Basic Structure

In this section, will be given a short explanation about the music structure, so the reader of this thesis can understand what are the music elements, which are going to be represented in the visuals. This section is an introduction to those music elements, and what elements were chosen for this thesis.

Music is a form of art expressed by sound and silence. And has so many different characteristics, which enables the distinctive emotions intrinsic in music. Musicians can express their feelings through music, which then become a source of inspiration for other people. One special characteristic that makes music so special is its level of intensity, which is always changing, growing as it develops over time. This is possible because music is comprised of several elements: dynamics, tonality, timbre, tempo, duration and pitch. All these elements are important and only musicians understand all of them. People without any experience in music, can't understand all these elements.

3.1. Dynamics

This is probably the element better understood by the majority of people because represents the volume of music. The volume of music is noticed by everybody, because doesn't require a musical trained hear to understand this element. People can immediately recognize a "crescendo", i.e. the volume is low and then increases, or even a staccato note, this means, when they hear an abrupt sound.

In my point of view, it is the main music element to work with in music visualization, because people who can't listen will immediately identify the dynamics of a music piece. This is why this element is chosen to translate it into visualization, which will be explained in section 4.1.

3.2. Tonality

As it was described previously (in section 2.1.2.4. Visual and Aural- Visualization of Harmony in Music with Colour), tonality can be major or minor depending on the relationship between pitches of music. In western music, all major tonalities have the same intentions, in other words, the major tones are related to joyful feelings. In the other hand, minor tonalities also have the same intension: unhappy feelings.

In (Klemenc, Ciuha, Subelj, 2011) it is said that major tonality represents cheerful mood, and minor tonality the opposite. This thesis uses this knowledge in order to depict the feelings/atmosphere of music through visual means.

People can recognize different moods within music and those different moods represent different tonalities. Which is why this was another music element to map into visuals, because people have different perceptions of a certain music when the tonality is different.

3.3. Timbre

Timbre is the sound or tone, which differentiates distinctive kinds of sound production, such as voices, and musical instruments, wind instruments, and percussion instruments. In other words, timbre is what makes one musical sound different from another. This music element is responsible for giving the information of the different sound of the instruments.

It could be used to trigger different visual for each instrument, but that could not be a good solution depending on the type of music. For instance, if it is decided to trigger animations for each instrument, imagine that one instrument has only a small part in a certain music piece, then it will trigger little animations. In this project, all the visuals were tested with the same kind of instrument (mandolin), this is why the visuals would not be surprising or introduce novelty.

3.4. Tempo and Duration

These two music elements are deeply connected, because tempo is the beats per second in a music piece, is the speed or pace of a given piece and duration is the length of a musical note according to the tempo. These two elements are also not used in the visuals, because it is extremely difficult to know in a real-time environment the beats per second of a music, because it can change, and also the duration, because musical notes can have different duration lengths.

3.5. Pitch

This music element is a perceptual property that allows the ordering of sounds on a frequency related scale. Pitches are compared as higher or lower in the sense associated to musical melodies. This is another music element that can be detected in a live performance and depicted in visuals.

There are several real-time pitch detection algorithms created nowadays. Performance environments can contain a lot of noise, therefore it would be good if pitch detection algorithms could tolerate noise, because it is a problem in interaction music. There are a few requirements that should be considered in these algorithms, such as:

frequency resolution of at least semi-tons, including the correct octave; timely recognition and quality of instantaneous pitch, to convert into a pitch in real time.

In (Cuadra, Master, Sapp, 2001) they talk about four algorithms: Harmonic Product Spectrum (HPS), Cepstrum-Biased HPS (CBHPS), Maximum Likelihood (ML) and the Weighted Autocorrelation Function (WACF).

HPS:

It is based on the Discrete Fourier Transform (DFT, is a way to decompose a signal into its constituent frequencies) technique, although it loses resolution at lower frequencies. Its primary drawback is the need to enhance low frequency resolution with zero padding (extending a signal or spectrum with zeros) of the signal before transforming, so that the spectrum can be interpolated to the nearest semi-tone. Also, wastes computational power because interpolates the high frequencies, but it runs very well in real-time (Cuadra, Master, Sapp, 2001).

CBHPS:

It is also based on the DFT (as the HPS). It is the most known pitch-tracking method in speech analysis technique, additionally is good for multi-pitch detection, since it robustly handles noise and pitch errors (Cuadra, Master, Sapp, 2001).

ML:

This algorithm behaves well at low frequencies, because it doesn't have the necessity of zero padding. But it is only effective if the input source is a fixed tuning like keyboard or woodwind instruments, string instruments or voice are not good because can produce non-discrete pitches, like vibrato (Cuadra, Master, Sapp, 2001).

WACF:

This algorithm is accurate and computationally inexpensive time-domain technique, although its resolution is limited by the sampling frequency (Cuadra, Master, Sapp, 2001).

For this project, it was decided not to map some music elements (duration, tempo, timbre) because they are not that important for visuals. For instance, if certain music contains a lot of short notes, then the duration would be always the same. This means that the visualization would be the same throughout all music pieces. And we need dynamic elements to have flexible and different kind of visualizations.

Additionally, those elements (duration and tempo) are very difficult to detect in a live performance as it was explained before. They can’t be used in a live music performance due to its specific properties, this means that it is very hard to know what is the beat per second in a live concert as the duration of the musical notes, because this depends on the musician interpretation. Each music professional has his own way of understanding pieces and playing it, and when they play these music elements (tempo and duration) will be always different. Although, tonality and dynamics are elements that will not change, therefore they are easier to detect in a live music performance.

Finally, these music visualizations of this thesis were used in a mandolin orchestra live performances, so the only kind of sound existent is mandolins, this is why it makes no sense detecting the timbre element.

4. Methodology

Based on the music elements stated on the last section, now it is going to be explained how the music mapping into visualization was created. In other words, it is explicated how the dynamics (volume) and tonality are going to be converted into visuals. What kind of parameters they will change in the visuals during the live music performance, and if that mapping is really functional, this means, if people can relate music with the visuals.

4.1. Music Mapping into Visualization

Dynamics (or volume) and tonality were the two music elements chosen (as it was explained in section 3) and are going to be used as parameters for the graphic animations. The visualization must be reacting to volume and tonality of music in real-time. During all the process, several mapping functions were created with the goal of reaching the best result. In other words, this thesis tries to find the best relationship between music and visualization, which people can "see" music through visualization. For start it was tested several hypotheses (see table 1):

Music Element	Visualization Parameter
Pitch	Position
Volume	Color Intensity
Volume	Position in the Y-axis

Table 1 – Phase 1 of music mapping into visual

But these relationships were not giving the desired feedback, so the obtained result wasn't the expected. In section 5.1. it is explained why it wasn't working in the desired mode. Although, the relation between volume and position in the Y-axis seemed to work, because it helped the audience to identify those high moments, of high peaks in the live music. Then was tried (see table 2):

Music Element	Visualization Parameter
Tonality	Color
Volume	Position in the Y-axis
Volume	Size

Table 2 - Second phase of music mapping into visualization

These relationships were the result of the feedback from the previous live performance with music visualization. The audience felt that was missing some color, and the music should have more reactions with music so they could understand that the visualization was really reacting with music. That is why it was decided to play with size, because it's something visible, which anyone can relate the size of an object with the volume of the music immediately. Also if the color could change with the music mood, it would make this experience more engaging as it was mentioned in the related works (see table 3).

Summing, this thesis tries to detect the music tonality in order to have the knowledge of the music mood, then it is used warm colors for cheerful mood, and cold colors for tragic feelings of music. For size, all the visuals are small whenever the volume is low, and then increases with the volume. To finish, the position of the visual is also according to the volume, this means that, if the volume is low then the position of the visual is at the bottom, and if it gets higher, the visual position will be at the top. Furthermore, some visuals have random movement, because if the volume is constant, then the visualization will not have changes, this is why was decided to use random movement for some visuals, furthermore the speed of this movement is reacting with the volume.

Music Element	Visualization Parameter
Tonality	Color
Volume	Position in the Y-axis
Volume	Size
Volume	Speed of movement

Table 3 - Last phase of music mapping into visualization

4.2. Implementation

Last section was discussed the mapping functions that are going to be implemented, therefore, based on those mapping functions, the visualizations will work accordingly. In this section, is explicated how those functions are going to be implemented. Additionally, how they were implemented and all problems found, as well as solutions to fix those issues of the implementation of this project. This section is divided by software, hardware and visualizations in order to give an explanation of each of these implementation steps.

4.2.1. Software

There are several technologies available to make this project possible as it was mentioned in section 2.2. The choice of the software for music visualization was based on the specific requirements of reacting visuals in a real-time situation: quick responses to music, the visualization can't freeze and has be always changing according to what is happening in the music at the moment. Thus, software has to support these requirements, and have the possibility of making strong graphic animations.

Quartz Composer is Mac OS software, and is free for any Mac user (see section 2.3). There is a considerable number of documentation on the web (namely in Kineme[2] forum), which helps in the development process of any quartz composer user. Quartz composer is able to produce real time visualizations and can be used with other software to improve the results for e.g.: we can use Pure Data (section 2.2.2) with Quartz Composer. In Pure Data we can use the fiddle patch (which is a good patch to detect pitches in real-time) and then send those pitches to Quartz Composer using Osc Patch (which creates a communication port and then send the values from one process to another through UDP protocol). The problem with this is having two programs running at the same time, which will consume a lot of processing power. Quartz composer is a visual programming software that creates all kind of graphical animations. Due to the visual programming environment, it makes easier creating animations with a high level of quality for programmers with little experience.

For this thesis, was decided to use only quartz composer, because the music visualizations were made for an hour performances and sometimes the quartz composer crashed, which means that the visuals have to be optimized. Using Pure Data to detect pitches would consume more processing. Additionally, Adobe software (Illustrator CS5, Photoshop CS5) was used for image editing in order to create different and original images.

[2] http://kineme.net/

4.2.2. Hardware

For this project was used several hardware, each one with a specific task. Firstly, it was used a Mac OS 10.7.4 Pro, 2.3GHz Intel Core i5 in order to run Quartz Composer and develop the system. Secondly, was necessary to use microphones to capture real-time audio data, such as: volume, frequencies and sound spectrum. Nowadays, there are several kinds of microphones (omni-directional and directional) with different objectives. For testing purposes it was used the internal microphone (from the MacBook Pro) or the omni-directional Samsung microphone connected via USB.

In performances it was used the Samsung microphone, or the rented microphones from Company Mega Som Madeira[3] (2 omni-directional and 2 directional). Moreover, was necessary to find a way to connect several microphones into the computer. The solution was using an audio interface, which should have at least four XLR inputs, and a USB or Firewire output to connect into the computer and thus receive all sound data.

Another important hardware for this project is the projector, because all graphical animations were projected onto a surface. So the projector needs to have good quality, else, if the visuals have poor image quality, then the audience will have difficulties watching the visualizations, and understanding what they are seeing, thus they will loose their interest in the visualizations.

4.2.3. Visualizations

All visualizations were created in QC (Quartz Composer) like it was mentioned previously. The visuals represents the mental model of what the author imagines when listens to those music, additionally was decided to use abstract visuals so it could be "universal" and adequate for any kind of music. Using pictures would limit that specific music piece to those pictures, hence the choice of using abstract visuals. Abstract visuals are flexible, and can accompany any kind of music.

People have their own mental model of a certain music piece, and of course each mental model is different. But they also can relate their mental model with others mental model of what people imagine when they listen to music. The ultimate goal is that people can find a relationship between the thesis' author music visualizations and their mental model of a specific music piece. Afterwards their mental model can change due to the visualization, or it can remain the same if they don't find any connection between the visuals and what they imagine.

[3] http://www.megasommadeira.com/

Pitch detector

There are two different hypotheses in order to detect pitch in real-time: fiddle patch from Pure Data and Primary Frequency patch from QC. Fiddle patch from Pure Data (figure 62) is an object based o FFT algorithm that performs analysis of both volume and pitch, also can detect the partials and theirs respective volume of the input signal (Pure Data, 2012).

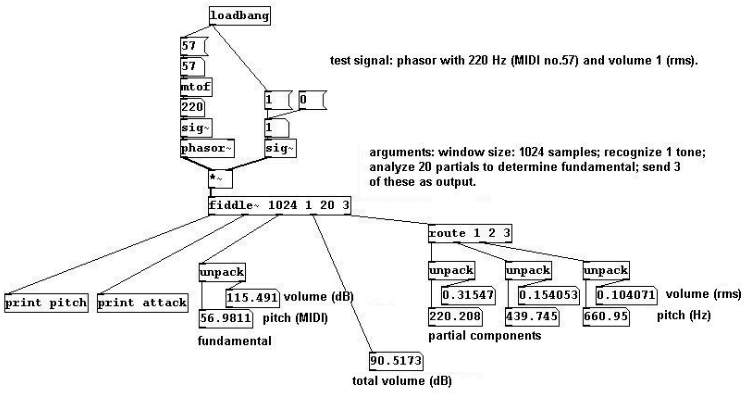


Figure 62 - Fiddle patch from Pure Data

The primary frequency patch from QC (figure 63) finds the primary frequency using either a simple maximum, or the Harmonic Product Spectrum

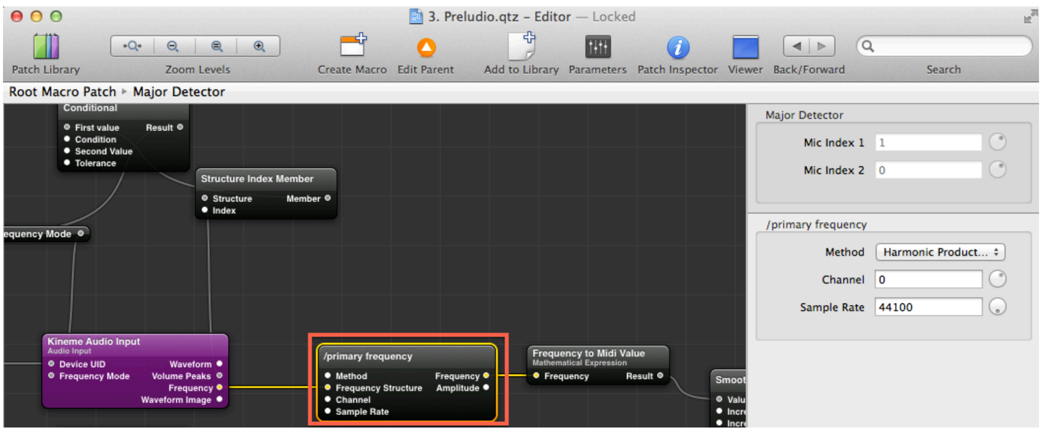


Figure 63 - Primary Frequency patch from Quartz Composer (red rectangle)

As it was mentioned before, for this project the primary frequency patch was the chosen hypotheses for this project, due to processing power, although the fiddle patch from Pure Data more accurate.

Major Detector

Having the knowledge of last section is possible to detect the tonality of music. As it was said earlier, tonality can be major or minor. It depends in the relationship of the frequencies that are being detected at the moment. That relationship is easier to detect working with MIDI values rather than with frequencies. So it was used a formula to convert the frequency values to MIDI values and then detect the music tonality or tone (figure 64).

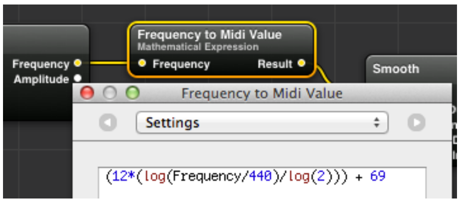


Figure 64 - Formula used to convert frequency to MIDI value

It is necessary to have the two MIDI values (it is necessary one microphone with two channels, or two microphones with one channel) and then see the difference of the detected frequencies. With this difference or interval between the frequencies we know if it is a major interval or minor. The value of the difference represents a major or minor interval. For instance, we have several major intervals, like a third major interval. In order to calculate this interval, the difference between the two MIDI values is four. In the other hand, if this difference is three, then is a third minor interval.

Mapping Function

The mapping function changed during the course of the development of this project in accordance with the different hypotheses. These hypotheses (discussed in section 4.1.) were tested in the visuals in order to know if they would be meaningful for the music visualization.

After knowing the Boolean value of the major detector, then the visual would have the chosen colors according to the Boolean value. In other words, there is a set of different colors for major tones, and another set of colors for minor tones. If it was used just one color for major tone, and another single color for minor tones, then the visual would be less interesting and people like colorful visuals that are always changing in a way they don't expect. Regarding to volume, it was used the following formula in order to convert from one scale to another (figure 65).

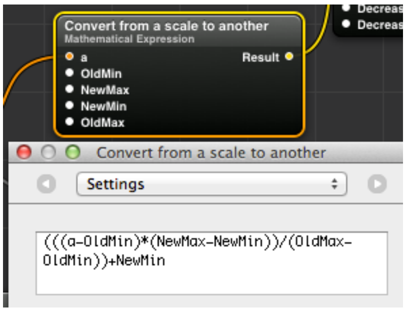


Figure 65 - Formula to convert from a scale to another

The volume has values from 0 to 1 (1 is a volume peak), and then it was necessary to translate the volume values to coordinate values, or the visuals size. So the “OldMin” and the “MaxMin” correspond to the 0 and 1, i.e., the minimum value of the volume scale and the maximum value. The “NewMin” and “NewMax” can be the new minimum value of the Y-axis position of the visual, and the new maximum value of the Y-axis position.

Later on, was decided to use the “OldMin” and “MaxMin” value as “MinTreshold” and “MaxTreshold”, because with the different microphones, the captured values are different and it reflects on the visualization. Moreover if the values are different, then the result won’t be the expected. Using this threshold, we can resolve this problem, for instance, if the maximum volume peak is 0.2, then we should change the maximum threshold to 0.2.

Video Mapping

One the challenges for this thesis was projecting the graphical animations onto a certain surface (e.g. circular surface). In order to solve this challenge, was used the Perspective patch from Kineme, which is capable of warping an image. And this image will be the background where we will project. For instance, section 5.2. is described a live performance which the music visualization was projected onto a big circular support. So in this case, the perspective patch was used to warp a circle, and then all the visuals were inside that circle.

In order to do the video mapping we have to follow these steps: Firstly, save the coordinates of the four corners into a plist file (figure 66). To save these coordinates, it is used a circle image on each corner, and then we adjust each corner until the corners are fixed in the desired place (figure 67).

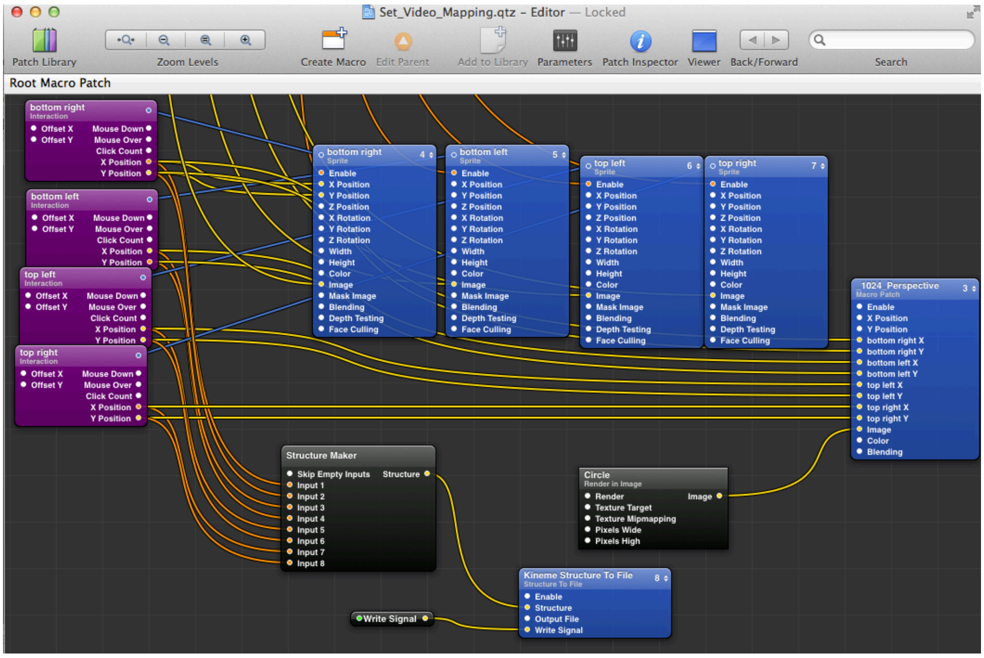


Figure 66 - Quartz Composer file to setup the video mapping

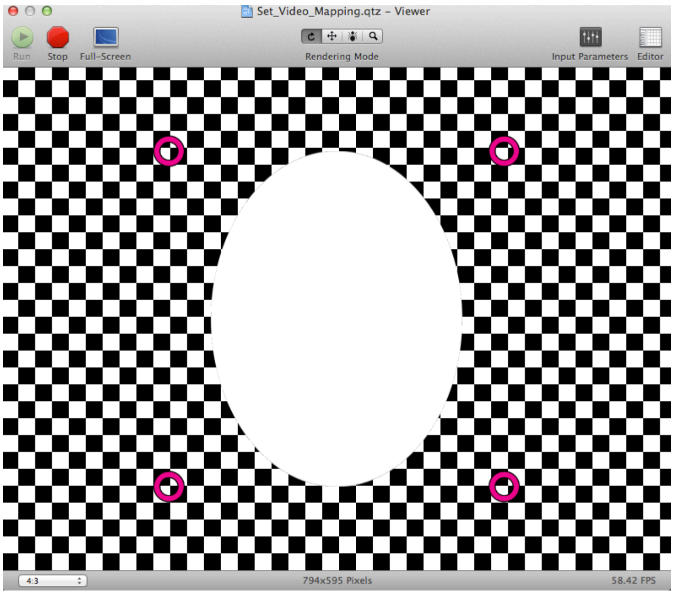


Figure 67 - Viewer window of the setting up the video mapping QC file

Figure 67 shows an example used for one of live performances settings, where the visuals were projected onto a circular surface. As you can see in figure 67, the circle is projected onto the surface, and then using the pink smaller circles we adjust the four corners until the big circle matches the circular surface. And then, all the visuals will be projected onto that circle. The next step is to save the coordinates of the four corners and then whenever it is loaded a visual, it should be loaded inside that circle with the saved coordinates.

Afterwards, there is another quartz composer file, which will load the coordinates values into the perspective patch, and then we have to publish the image parameter from the sprite patch (see figure 68) and then use this QC file as macro patch in order to load the desired animations (figure 69).

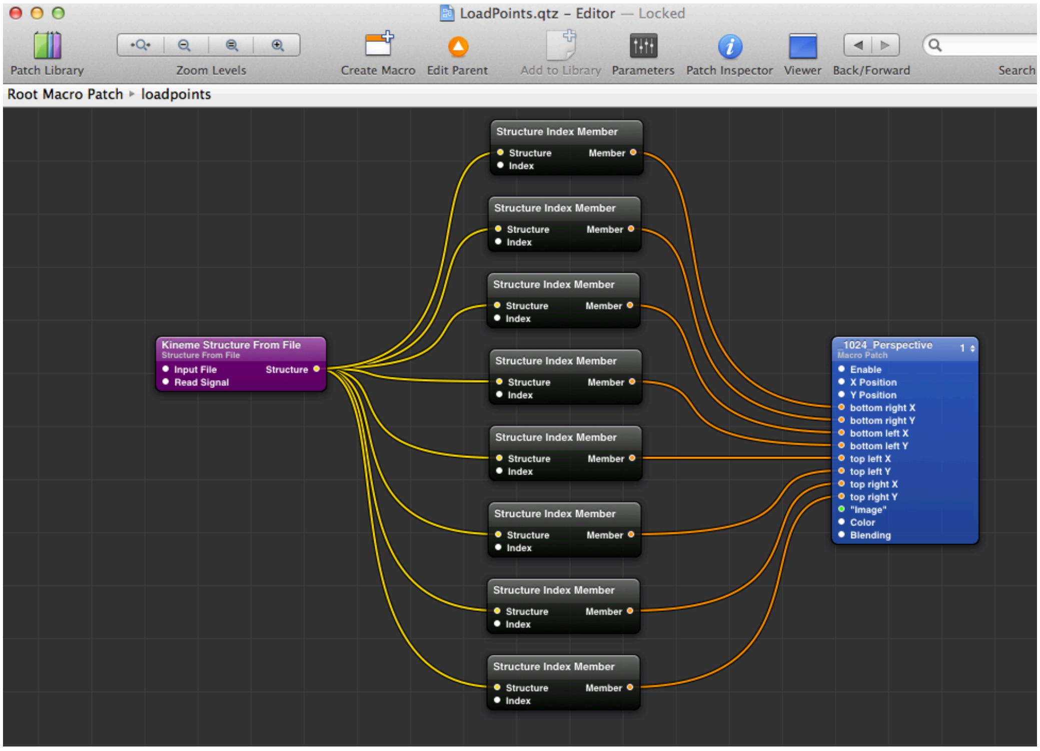


Figure 68 - QC file which loads the corners coordinates saved for the video mapping

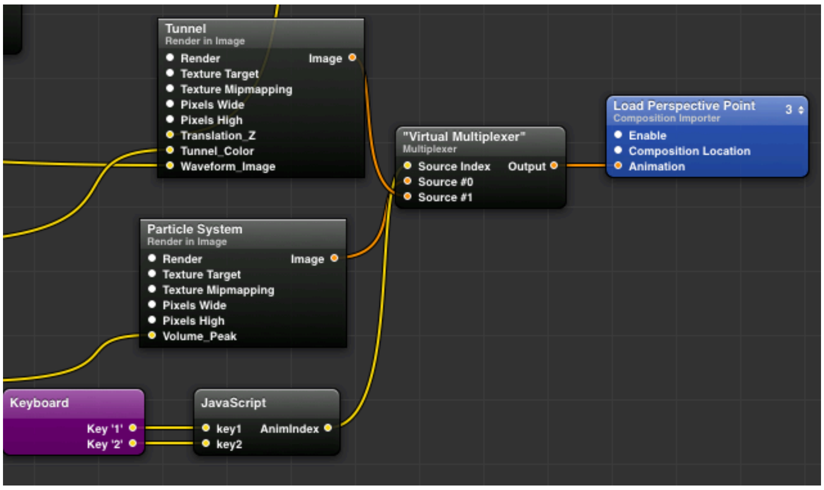


Figure 69 - Perspectives Points Macro patch with the image parameter published as Animation

Finally, connect all the animations into the published image parameter (which will be available in the “Load Perspective Points” macro patch) (see figure 69).

Visualizations examples

As it was explained previously, all animations used are connected into the macro patch "Load Perspective Points". In order to connect multiple visualizations it was used a "Multiplexer" patch, which will output one animation at the time (figure 76). The index of the chosen animation to be output is controlled by a key event (e.g. press key "1" to output animation in the first index of the "Multiplexer" patch).

The following graphical animations are some examples of the visuals used for all the live music performances with music visualization throughout the project. All of this music visualizer reflects the mapping functions between music and visuals (as it was discussed in section 4.1).

Figure 70 depicts a visualization used in the first live performance (see section 5.1 for further details about the performance), which the pitch is related with the position, in other words, when there is a high pitch, then the position of the animation will also higher (in the Y-axis and also Z-axis).

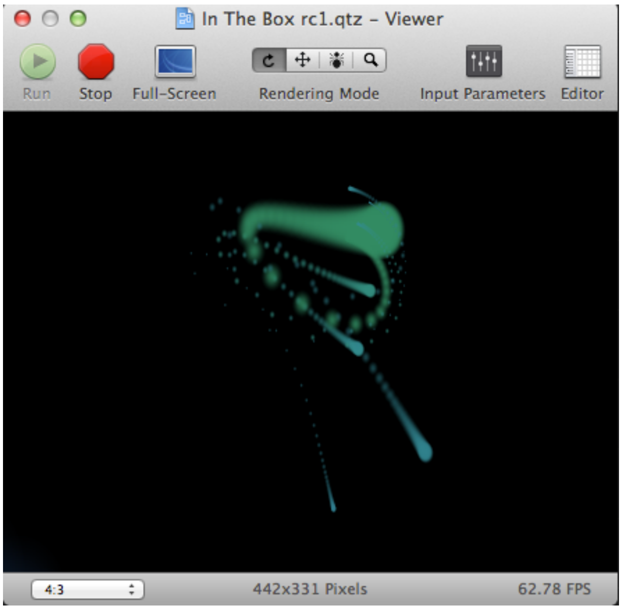


Figure 70 - Visualization Example from the first live performance

In figure 71 is also represents another visual used for the first live music concert, and in this visualization the volume was mapped into color intensity. So the color is more opaque with volume peaks, and transparent with low volume (as it was discussed in section 4.1).

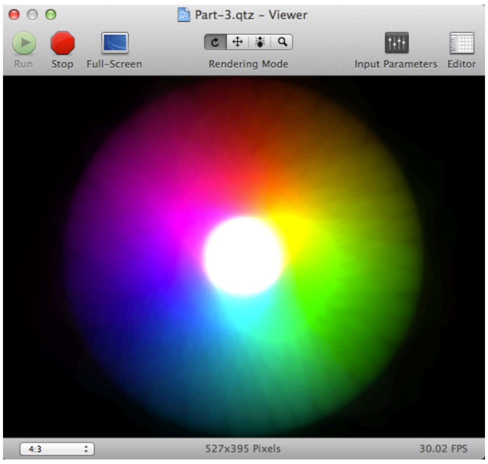


Figure 71 - Visualization Example from the first live performance

Figure 72 illustrates also another example from the first lice music performance, which was at University of Madeira. And in this last example, the volume controls the position of this colorful system particle, for example, in this figure we can see that the system particle is going down, so the volume is decreasing.

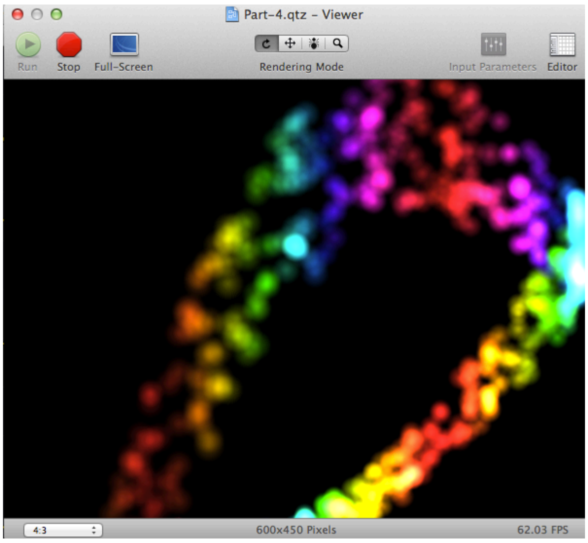


Figure 72 - Visualization Example from the first live performance

In figure 73 is exemplified one visual for the second live music performance (see section 5.2 for further details about the performance), which had a great feedback from the audience. They said that it was very clear to understand the volume through the cube. In this case the volume was mapped into size, and color to tonality (as it was discussed in section 4.1). This visualization is suitable for music pieces with frequent volume peaks, as it is visible with the height of the cube, therefore there are many changes in the visualization.

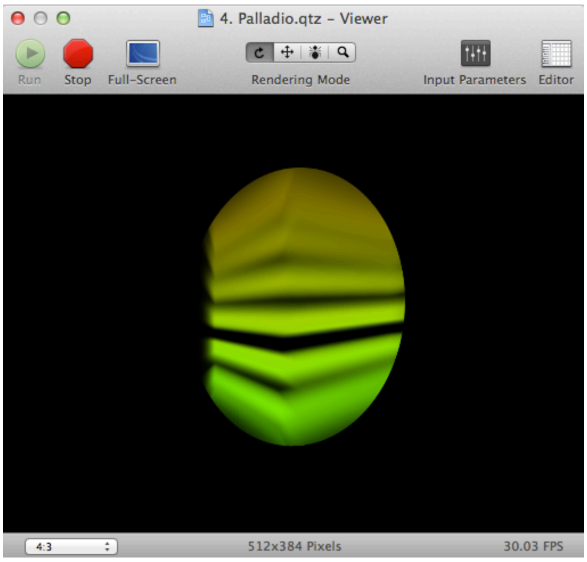


Figure 73 - Visualization Example from the second live performance

In figure 74 we can observe another example of the second concert. In this one, the visual has a particle system, which its colors change with the tonality, in this case it is a minor tonality, because the colors are dark. Furthermore, the visualization has lines that resemble the wings of a bird. The volume is portrayed by the movement of the "wings", this means, that if the volume increases the "bird" will spread its "wings".

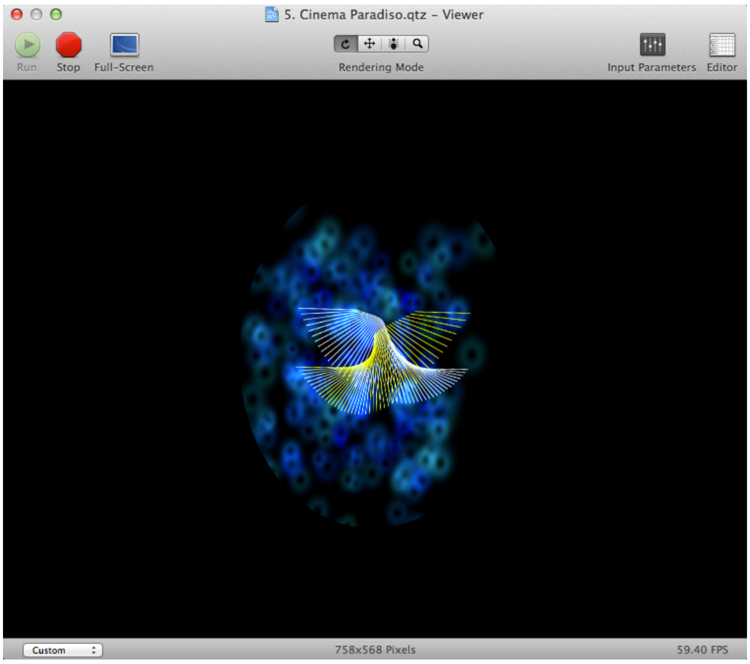


Figure 74 - Visualization Example from the second live performance

Figure 75 is an example, which is also a system particle with some superb explosion effects triggered according to volume peaks. Again the color changes according to tonality. This kind of visualization is efficient for music pieces which has sad feelings. In the author's opinion, these explosions effects recalls war, this is why there is this negative mood present in this visualization.

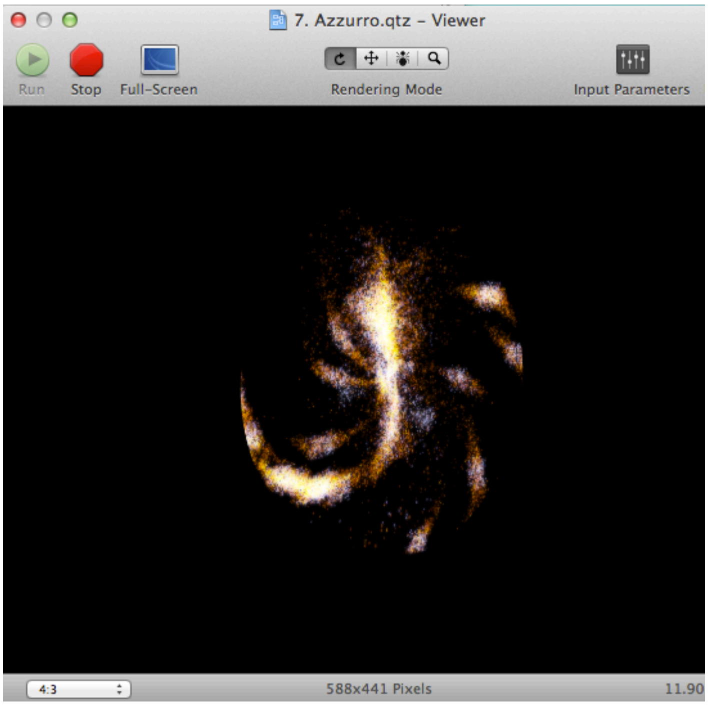


Figure 75 - Visualization Example from the second live performance

In figure 76 is represented one of the visuals used in the third live music concert performed by mandolins (see section 5.3 for further details about the performance). This effect reminds fire, and reacts to sound. In other words, the volume is shown by the speed of the movement. So, the fire gets bigger and uncontrolled when the volume is higher. This is another different mapping function added to music visualization, which was helpful in order to achieve the main goal of finding the appropriate mapping between music and visualization.

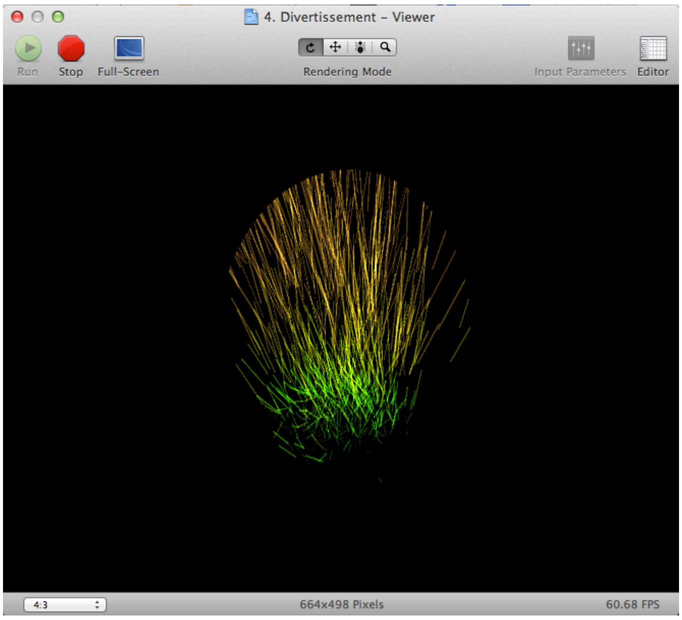


Figure 76 - Visualization Example from the third live performance

Figure 77 represents another simple visual reacting to music used for the third live concert. It was decided to try several different design types in order to know what is the best suitable kind of design for this situation. Therefore, this visualization is only comprised of bubbles, which size changes with volume, and color with the tonality.

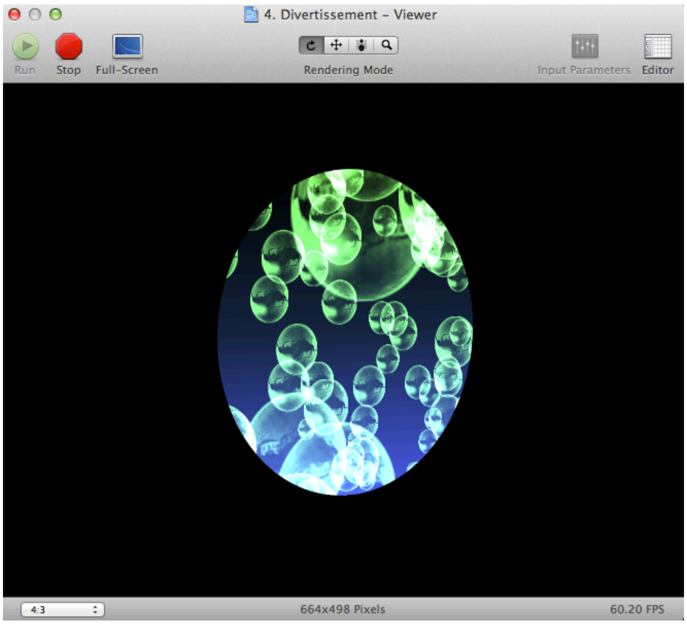


Figure 77 - Visualization Example from the third live performance

Figure 78 illustrates another kind of visualization, which depicts volume by the size of the trail of the stars. Stars with a random movement compose this visual, additionally, their trail increases when the volume increases, otherwise it is just a star without the trail effect.

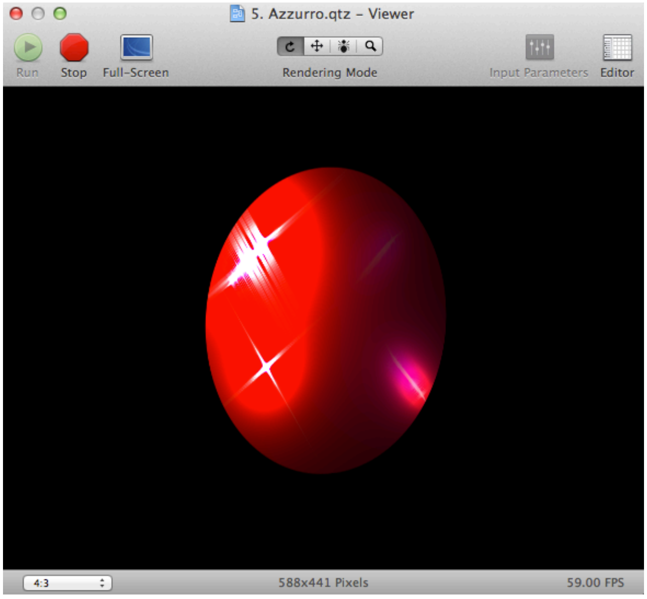


Figure 78 - Visualization Example from the third live performance

In figure 79 is illustrated a visualization with whirlwind movement, which is really effective to show crescendos and diminuendos, in other words, to show changes in volume (if it is increasing or decreasing). This whirlwind movement becomes stronger when the volume is stronger, else becomes weaker. Moreover, the color changes with the live music tone.

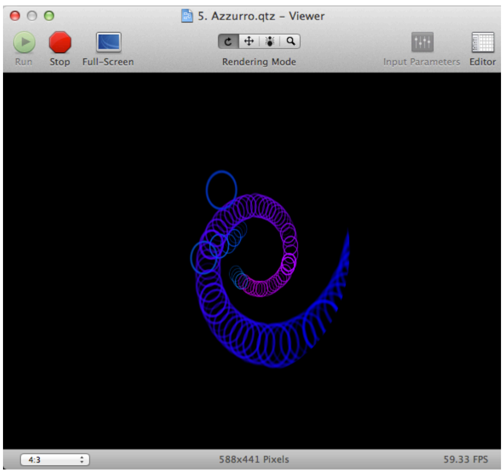


Figure 79 - Visualization Example from the third live performance

Also using the idea of figure 73, in figure 80 was decided to do another type of visualization with a cube. But this time with the final mapping functions agreed to use in this project. Then, some parameters of the cube are changed by music elements. For instance: the cube has a rotation movement, which is controlled by the volume (the rotation is quicker when the volume increases); there is also the size of the circles existent on all the faces of the cube that changes according to the volume (the size of the circle is bigger when the volume increases, or smaller when the volume decreases); finally there is the color which is controlled by the music tonality (major tones will be translated into light colors such as light green, in the other hand, minor tones will be represented by dark colors, such as purple).

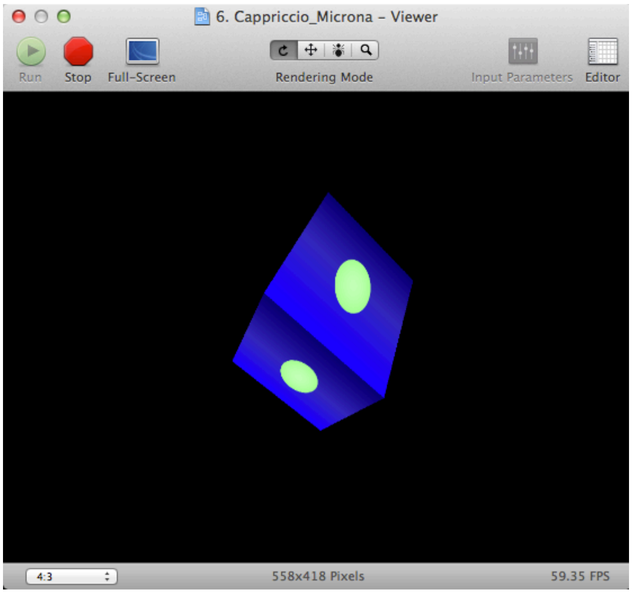


Figure 80 - Visualization Example from the third live performance

Figure 81 conveys another example which the audience liked the design (including the orchestra members). This visualization is composed by three particle system with different colors. These particle systems reacts with volume, so when there is a volume peak, the particle system will "explode", in other words, the particles will be less concentrated in one point, and will spread. So, with high volume peaks, we are going to see particles all over the animation of several colors, but in accordance with the music tonality.

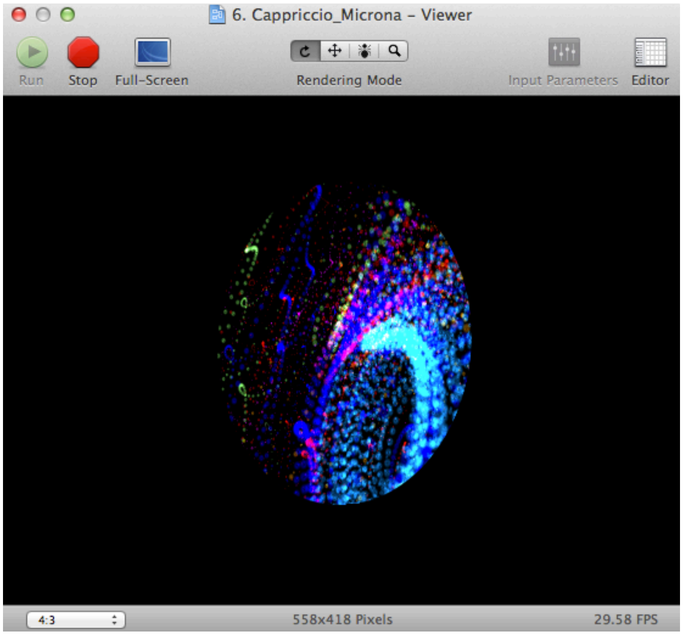


Figure 81 - Visualization Example from the third live performance

In figure 82 is possible to see one of the visuals used for the last live music performance, which was a musical drama (see section 5.4 for further details about the performance). This visual is again a particle system, which the particles are sabre swords. This visualization was created specially for the music piece Sabre Dance (by Aram Khachaturian), this is why it was chosen sabre swords for the visualization with the intention of designing a dance between the sabre swords. This graphic animation is reacting with volume (size of the particles are bigger, and there are more swords with volume peaks) and again the color is changing accordingly with the music tonality.

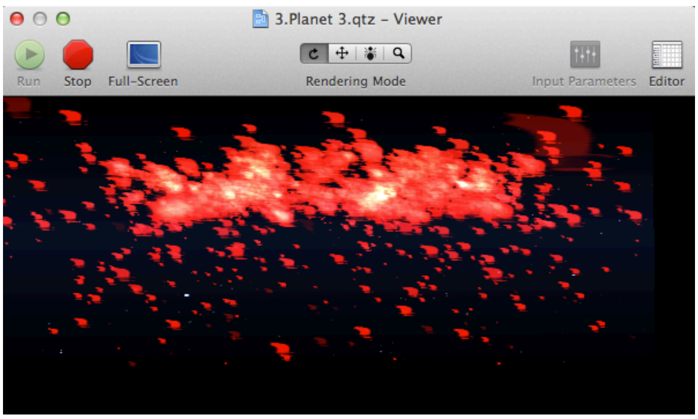


Figure 82 - Visualization Example from the fourth live performance

Figure 83 depicts again a particle system which reacts with volume. The first colorful particle system with pink, orange, and yellow tones in the bottom of the visual gets higher when volume increases. The other particle system only appears with high or low, or mid frequencies are detected. In other words, the sound spectrum is analyzed with the audio processor patch and then it is attributed a pink, golden and blue particle system to lows, middles, and highs respectively as you can observe in the next figure.

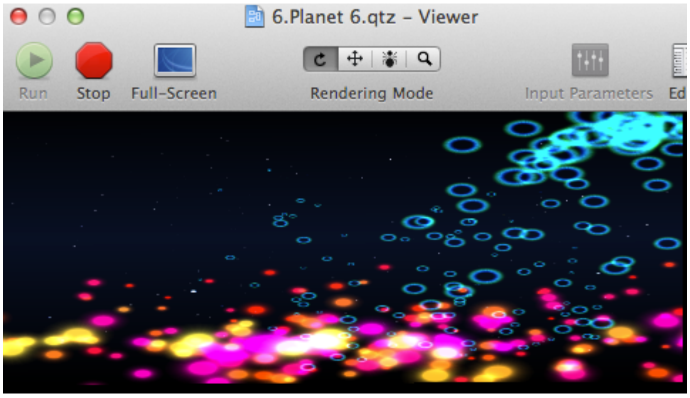


Figure 83 - Visualization Example from the fourth live performance

Figure 84 conveys another visualization used in the musical drama where it is present several particle systems. The particles of one of the particle system are musical notes (as is a show about the music world), and the particles of the other particle system are lenticular halos. The objective is to give a spatial look because the story of this drama is on space. Therefore, this visualization reacts to volume, in other words, with volume peaks, it will appear those four stars in the four corners and the size of the particles will be bigger. As usual, the color changes accordingly with the music tone.

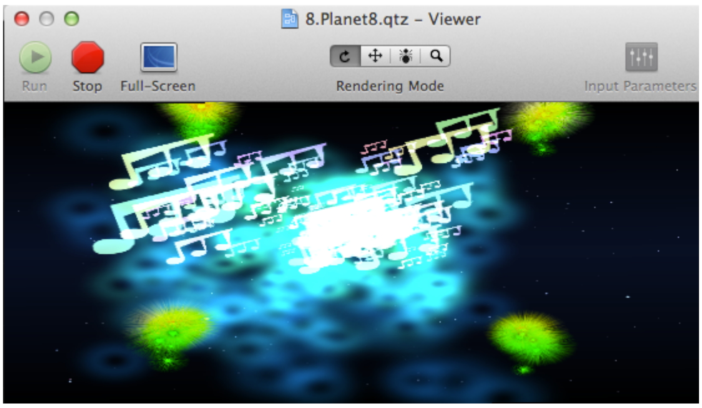


Figure 84 - Visualization Example from the fourth live performance

In this last example, is showed a visual animation that would fill the entire screen with many colors. Figure 85 illustrates the music visualization of the last scene from the musical drama. This was the happy ending scene, that's why it has so many colors and animations going on. You can observe in figure 85, that there are bubbles going up (the volume is conveyed with the size), there is also a system particle on top of the visual that only emerge with volume peaks, there are colorful circles going up and also the lines behind all these animations. Likewise, those lines move with the volume peaks.

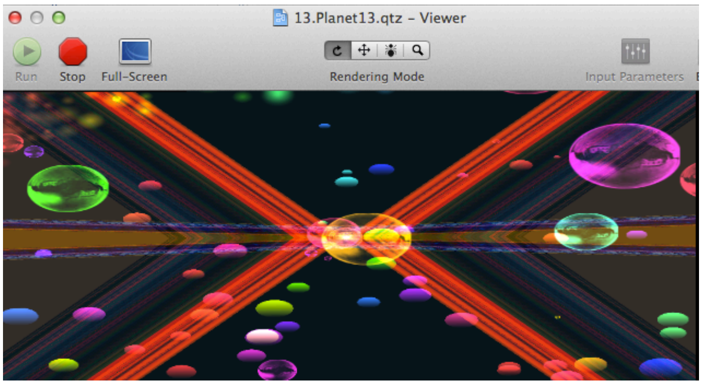


Figure 85 - Visualization Example from the fourth live performance

5. Live Performances with Music Visualization

All the music visualizations examples were used in different live music performances throughout this thesis development. All visual also had the goal of discovering several mapping between music and visuals in order to know what is the best way to depict music through visualization in a way that people can understand music simply by seeing the visualizations.

This section reports all live music performances with music visualizations occurred during the development of the project. There were four concerts, the first one was just with a small music piece, and then there were two more concerts at the Wine Institute with the Mandolin Orchestra of the Bureau of Artistic Education, and finally another one for a musical drama at John dos Passos.

5.1. Live Music Performance 1

This first concert occurred in the 13th of December at the bar of the University of Madeira, with the goal of introducing the theme music visualization to people. For this concert the author of the thesis played mandola accompanied by a MIDI piano (played by Menghua Lin). The first music piece played "Thais Meditation" (by Jules Massenet) without music visualization and then it was played "The Swan" (by Camille Saint-Saëns) with music visualization. As it was used a MIDI piano, it is possible to know with precision the pitches being played by the piano. Therefore, the visualizations were following the melody of the piano, in other words, when it was played a specific melody in the piano, then it would trigger different visualizations.

The goal for this live performance is to showcase the difference between a performance with and without live music visualization. In other words, show what music visualization in a live performance can offer to the audience, what they can feel in this kind of concert, additionally is necessary "educate" the audience for what is this kind of technology, and try to improve the audience live performance experience. The feedback from the audience was positive, because they like it, and also said that it was really calm and relaxing.

Summarizing, it was a good starting point because the audience like it and they said that it was more entertaining with the visuals. There was a software problem because the code had a bug, so the audience didn't see all the visuals prepared for that presentation. Additionally, it is necessary to find new mapping between music and visualization because people paid more attention to the visuals than the music, which means they liked the visuals but they didn't find a relationship (in this concert was used the mapping functions of table 1, from section 4.1). They only said that it was relaxing and beautiful, but they didn't say anything about the music. And if the mapping were successful, then they would say that they liked hearing the music with the visual, that they were deeply connected with music through the visuals, and this wasn't the obtained feedback from the audience.

5.2. Live Music Performance 2

This live performance took place in the 30th of March at the Wine Institute and it was used the mapping functions of table 2 from section 4.1 (see figure 86). The project was proposed to the mandolin orchestra of Bureau of Artistic Education conductor (Norberto Gonçalves da Cruz).

Figure 86 - Live Music Concert with Music Visualization

The mandolin orchestra (which the thesis author plays) has monthly concerts at the Wine Institute with the objective of promote Madeira Wine and embroidery through music events. Furthermore, all these monthly concerts have a theme, and for the March theme was "travelling around the world", the orchestra played music pieces from Italy, Japan and Argentina.

This project was integrated in the performance in order to achieve the challenge of connecting visuals with the Madeira Wine Institute heritage component. For start there is a brief presentation about Madeira Wine and a wine tasting session and then the audience goes to other room were the orchestra plays. The first part of the concert consisted in two duet music pieces for mandolin and accordion and the second part there were five pieces performed by the Mandolin Orchestra.

The goal for this performance was to enhance music properties (section 3) and the performance experience. Also was required to project the visuals onto a big circular surface (4 meters diameter), which contains a huge special embroidery that conveys the

Portuguese “Age of Discovery”. A person controlled the animations projected onto that circular surface, this means that the person was responsible for the controlling steps: opening the animation file of each music piece, and triggering different episodes of the visualizations using the keyboard keys.

Again in this concert occurred a bug in the video mapping quartz composer file with the interaction patch. Instead of saving the values of the coordinates of the four corners, it saved “inf” values. The solution found was erasing and disconnecting all connections between the interaction patches and the perspective patch, and insert the value for the four corners, and finally connect all over again. Other problem was with the microphones used in the concert, they were only available one hour before, so there was little time to test with those different microphones. The problem of using different microphones is the different values detected in the live performance. And the visuals weren’t flexible enough to adapt to those different values, therefore the result of the visualizations wasn’t the expected. Additionally, there was one visualization that crashed during the live performance and then the computer became really slow for a short period of time. The solution for this one is to optimize this visualization in order to run smoothly next time.

The feedback was again positive, the audience was very excited by the fusion of wine tasting, mandolin music, and interactive technology. The audience commented that the visuals made the performance more interesting and appealing, as they had a better understanding of musical feelings. They become more connected with music, which is very difficult when there is not music visualization. The audience reported that they had an intensive connection with performance through the visualizations, moreover they commented that the visuals were attracting and eye-catching.

Some paid more attention to the visuals while being able to follow the orchestra and music. This suggests that the mental model created by visuals of the live performance somehow corresponded to their previous mental model created without music visualization, and there is perceivable connection between the music performance and the visualization.

The spectators said that visualizations should have transition effects between the visualization and more colors. Finally, the conductor liked the outcome and wants to continue with this project, he mentioned that everyone is more involved with music and the musicians try to do their best in order to enrich all this new experience.

5.3. Live Music Visualization 3

This was the last concert of this year in Wine Institute by Madeira Mandolin, which occurred in the 26th of June (figure 87). The motive was to resume everything that happened all year, and play some special music pieces.

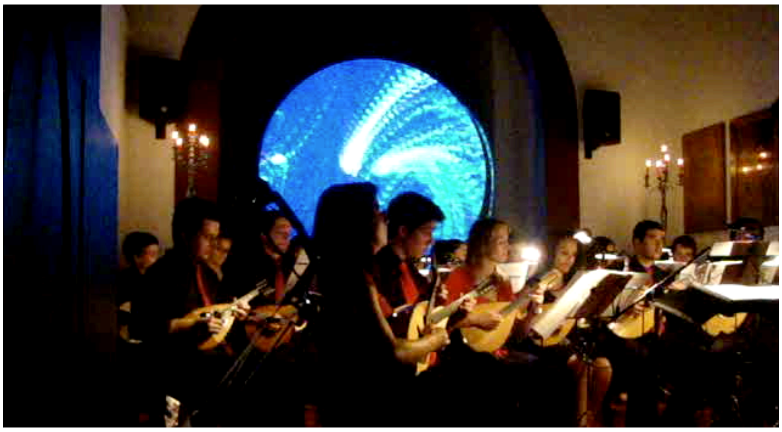


Figure 87 - Live Music Performance with Music Visualization

It first started with two music pieces without music visualization, the following music pieces had visuals, and for each one there was more than one visual. Again was necessary somebody to be at the computer and change the graphic animation in the right moments. This person, also need to have some music experience and know the music pieces that are going to be played at the concert. In this case we can see the computer as a musical instrument, so the person with it has to follow the music score or know well the music in order to change for the right visual. Between each visual there is a fading effect (didn't have last concert) in order make it smoother.

Also, for this concert was used an AppleScript in order to launch the animations in full screen mode in the secondary screen. Because in March concert, was necessary to put the quartz composer viewer window in the secondary screen, and then choose the full screen rendering mode. Additionally, detecting different values from different microphones isn't a problem like it was in last concert, because it was used a threshold to define what is the maximum value detected by the microphones. However, this time the microphones used were ours. So there was the chance to test at home with the microphones before the concert.

With this concert, there was the attempt of improving visualizations since last concert through new mapping functions (table 3 of section 4.1.), and test different solutions of last performance. This concert was better than the other, because the author of this thesis had the experience of the other concert, the only problem this time was with interaction patch of the video mapping quartz composer file. However, tests at Wine Institute started earlier, and there was time to fix the bug.

The feedback was better than the previous one, there was one person from the audience that said that didn't like this kind of music, but with the music visualization it is completely different, as it creates this strong and intensive connection between the live performance and the viewer. Moreover, some people said that only came to the concert because they knew that this concert had music visualizations. So, with music visualizations the concert becomes more interesting even for people who don't like some different types of music.

5.4. Live Music Visualization 4

This one was a special performance, because this is a project of the undergraduate musical science students that took place at John dos Passos in the 14th of July. And their project is a musical drama, which they have to prepare a show all by themselves. They are in charge of: music, production, scenery, etc. One of them watched the concert in March and found interesting the idea of music visualization, and wanted the music visualizations to be part of their show. There was a meeting with all the elements ahead of that project in order to show what music visualization is, and what is possible to do. Also it was discussed what they wanted for their concert.

For this performance, there are several kind of instruments, not only mandolins like the previous concerts, but violins, wind instruments, drums, piano, voices… it is a big dimension and different from all of the previous concerts (figure 88).

Figure 88 – Musical Drama with Music Visualization

This musical drama piece is based on the "Little Prince" by Saint Exupery story, and they have several planets representing different scenes related to music (since they are music students). The goal for this project is to enhance the musical drama piece and improve the scenery. As drawback, the visuals won't be noticed all the time, as it is a drama piece, people pay attention to the actors and only in the musical scenes they'll look for the visuals. For each planet there was a different visualization so the audience can understand that they are in a different planet.

For this project was used different kind of microphones, because they needed microphones for the orchestra, for the singers, and the choir. Mega Som Madeira again was responsible for the microphones and sound of the show. It was used a jack cable from the audio table responsible for sending the audio at the stage.

With this musical drama, more people get to know about music visualization. In this case, the majority of the cases, the visuals resulted as an extension of the scenery of stage. The audience found that it very interesting, because the visuals helped materializing the imagined world of this drama piece. The musical parts, they noticed that the visuals were reacting to the music at moment, and they felt involved with the piece. In this case, instead of only instrumental, there is also singers and helps people imagine the intention of the story. Though, with visualizations they have a better perception of the story, concluding, music visualization used in this kind of situation, not only help people understand the feeling intrinsic in music, but the fictional world of the drama scenery becomes "real".

6. Results/ Discussion

After the live performances with music visualization it was decided to conduct an online survey in order to prove the correctness of the formulated hypotheses for this theme. The hypotheses are: Does people with music background have a better perception of music feelings because they might already know the music and its story? Does people without music background feels the same as people with music background listening the same music piece? Is it different having music background when the audience creates their mental model of the same music piece? People can really notice that visualization is reacting with music (with and without music background)? And if they notice, which music elements they think that are in the visuals?

The online survey has three goals in order to respond to those previous questions: know what people feel listening to a certain music piece; know if people really realized that the visuals were reacting with music, and which music elements are reacting; lastly, know what people feel about the visualizations and what they envision (specially people with design experience)

In attachment A we can see the questions of the survey. The survey was also available in Portuguese and English in order to have more responses. It was obtained the total of 96 answered surveys. In order to answer the survey, it is necessary people to answer which music elements they think is embedded in the visuals. The possible music elements are: Volume, Pitch, Tonality, and Duration. These music terms might be confusing for people who aren't expert in music. So, in order to clarify novices in music, there is an audio example for each one, therefore novices can minimally understand these terms. For volume, it was used an audio which plays a crescendo, i.e., the sound increases so the test subjects can understand that volume can be increased or decreased. For pitch, it's played a sound of two different pitches, and the second pitch is higher than the second, therefore test users can understand that there are different pitches, and that they can be higher or louder. For tonality, it is played two different sounds, which are two different chords. Tonality can be major or minor, so it was played the same sound, but the first example was a major tone, and the second was a minor tone. Lastly, for the duration, was played two different sounds, and one had a bigger duration than the other, in other words, the second has a longer time length.

The questions from 3 to 10, are repeated three more times, for other music pieces, therefore it's possible to make some comparisons. The surveys were made using Google docs, and then it was embedded in a webpage along the video and audio files necessary for the survey. In order to analyze the surveys it was used the software Pentaho data integration (Kettle)[4], which gives the possibility of extract and transform huge amount of data. It's very good to integrate data of several sources and then obtain the desired data to analyze. In this case, it was created separated excel files in order to create different charts for each goal. Kettle has an intuitive, graphical, drag and drop design environment (like Quartz Composer).

It was exported the survey results as excel files and then it was merged both files (Portuguese and English). The answers of the Portuguese[5] and English[6] surveys are online. As it was expected, there were more Portuguese answered questions than English. The link for the survey was shared by email and Facebook, therefore we can have a bigger number of answered surveys. It was needed a considerable number of answered surveys so we can have a better validation for the hypotheses which we want to confirm through this survey.

6.1. First Goal

Analyzing the first case, the next figure (figure 89) illustrates the kettle file that exports two excel files (one file is people without music background and the other people with music background). As you can see, the first thing to do is to import the excel file (the one with all the answers). Then it was decided to change the name of the columns, because the current name was the question, so it was decided to use keywords instead of the questions for the name of the columns. After this, we just need to filter all rows by the people with music feedback (which the answer for question 1 is "No") and then export as separate excel files.

[4]http://kettle.pentaho.com/

[5]https://docs.google.com/spreadsheet/pub?key=0AnuS3BPJCcgAdE9wRXg4Sng4UDRNVmQ2by1FcHhMV1E&output=html

[6]https://docs.google.com/spreadsheet/pub?key=0AnuS3BPJCcgAdERxdWpJcGd1UnlvR2pNR1ZPal9wZmc&output=html

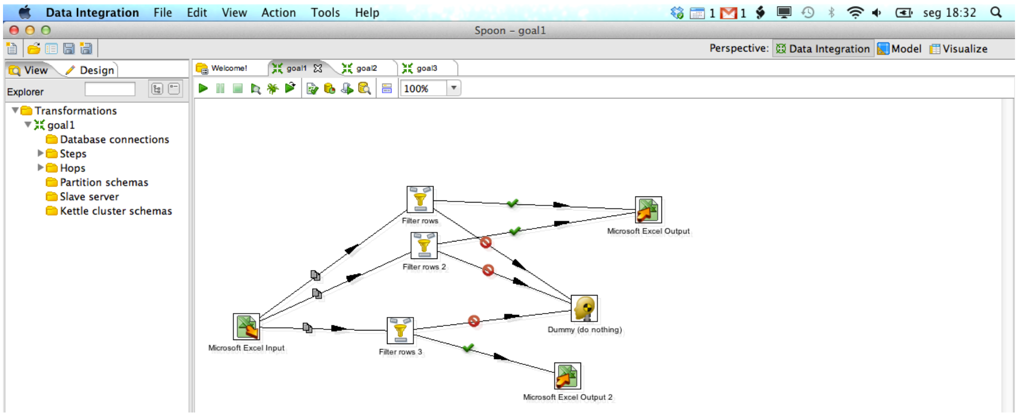


Figure 89 - Kettle file for the first goal

In figure 90, we can compare what people with and without music background feel listening to Cappricio Microna and Azzurro B. The first thing we can see is that the majority felt the same (energetic feeling) when listening to that music piece segment (Cappricio Microna). Still, we need to know what is the cause that provokes different feelings to people when listening to a specific music piece. Evidently, when the composer is working on a music piece, he has one intention, and the music has to be good enough so people can understand his feeling embedded in that music. Anyway, when people listen to a music piece, they have their own interpretation, they create their personal mental model of that music, and that is variable depending on their personal experiences. This is the reason why people sometimes feel differently when they listen to a music piece, and we can observe that in figure 90 and 91.

We can also conclude that people with music background have the capacity of interpreting music in a different way, because we can observe that there are several different feelings. For instance, in figure 90 people who listened Cappricio Microna, 48% felt an energetic feeling, almost half of the people. The other half is comprised of other feelings, such as: anguish, fear and expectation. So people who have better knowledge about music have more diversified kinds of feelings related to their interpretations as mentioned previously. Meanwhile, about 50% of novices in music felt an energetic feeling (figure 91), and almost 30% felt another kind of feelings. This is another prove for the previous conclusion, people have their own interpretation or way of analyzing stuff due to their personal experience.

This conclusion is important because when we are creating music visualization, we have in mind that everybody thinks differently about a certain music piece, even ourselves. But the important task of creating the music visualization is to know what is the feeling or interpretation from the majority of people. That is why, it is also important know well the music pieces and study them hard. In this case, we can see that despite the fact people are proficient in music or not, the majority (almost half of people) felt an energetic feeling when listening to Cappricion Microna segment, as for Azzuro B, exactly half of people felt sadness.

In a nutshell, the proposed hypothesis was: having a background in music enables people to feel music more intensively. This is true, because as it was discussed before, people with more knowledge in music felt more kinds of emotions due to the fact that they have a bigger connection to music than other people without that level of experience in music. This more intensive connection with music offers them a bigger universe, which allows those people the possibility of having other kind of feelings. When creating the music visualization, the creator have to really know what is the emerging feeling of that music piece and try to map it through the visualization. By doing this, the majority of the audience will be able to relate the visuals with their mental model created before the concert. Another interesting fact that can also happen, is that before the audience see the music visualization, they will have their mental model, their own idea of that music piece, and after the live performance with music visualization they can maintain their mental model, or they can change or even adapt to the mental model of the music visualization created in case if they were able to connect with it. Because, this will be a new personal experience for each one who goes to a live concert with music visualization, and if the visuals are good and really associated with music, then they can now remember those music pieces with the visuals created for that performance.

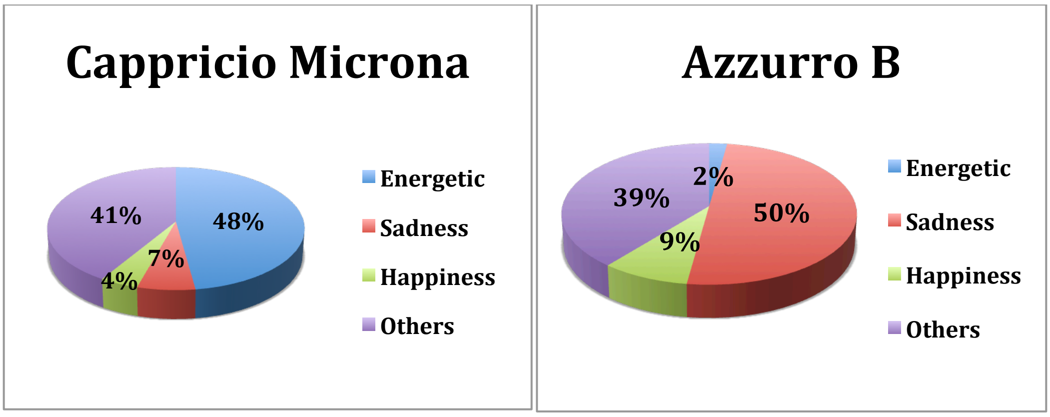


Figure 90 – Feelings of people with music background

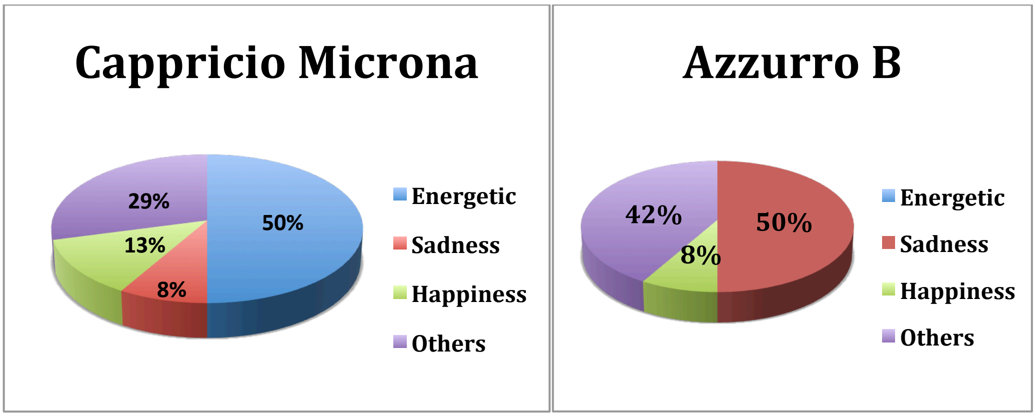


Figure 91 – Feelings of people without music background

6.2. Second Goal

For the second goal it was used again kettle software for the same purposes as the last kettle file mentioned. The goal here is to know if people realize that the visuals are reacting with music. And again it was separated people with music background and people without music background.

Figure 92 represents the chart where we can see the number of people with music background that noticed that the visuals were reacting with music. We can observe that the majority of people noticed that the visuals were reacting with music, principally in Cappricion Microna, where 97,83% of the user test noticed. Some people thought that visualization wasn't reacting with music, maybe that is due to the fact that the mapping function between music elements and visuals isn't good enough. An interesting fact is that the third music visualization wasn't reacting with music, but the majority thought that the visual was reacting. In figure 93, we can see that people without music background didn't noticed the visuals reacting as much as people with music feedback.

The music piece Azzurro B, was the one which the majority of the study users detected the mapping between music and visuals (75% of people). Anyway, far all music pieces over half of people without music background noticed that the animations were reacting with the visuals (figure 93). One possible reason for this smaller number of test users noticing the visuals reacting might be due to the fact, that they are novices in the music area, this means that they don't have the same music knowledge as musicians for example. These people don't now what is tonality, timbre, etc. This is why, in my opinion people only will notice the volume in the visuals. However, they might also perceive the music tonality, because the music tonality characterizes the feeling that the music composer is feeling while writing that piece. In other words, different episodes of a music piece have different tonalities, but that is not the only factor that characterizes an episode of the music piece, the rhythm also is another crucial factor for the creation of different episodes inside the music. So the rhythm or the harmony in conjunction with the tonality will form those different episodes inherent to a music piece. But in this project, it is only possible to detect the tonality of the music. But, there is one person controlling the animations, that person knows all the music pieces and then changes the animation in the correct music episodes. As it was explained before, tonality by itself is not enough to detect those music episodes. This is why it was decided to have one person to change the animations pressing a key of the keyboard.

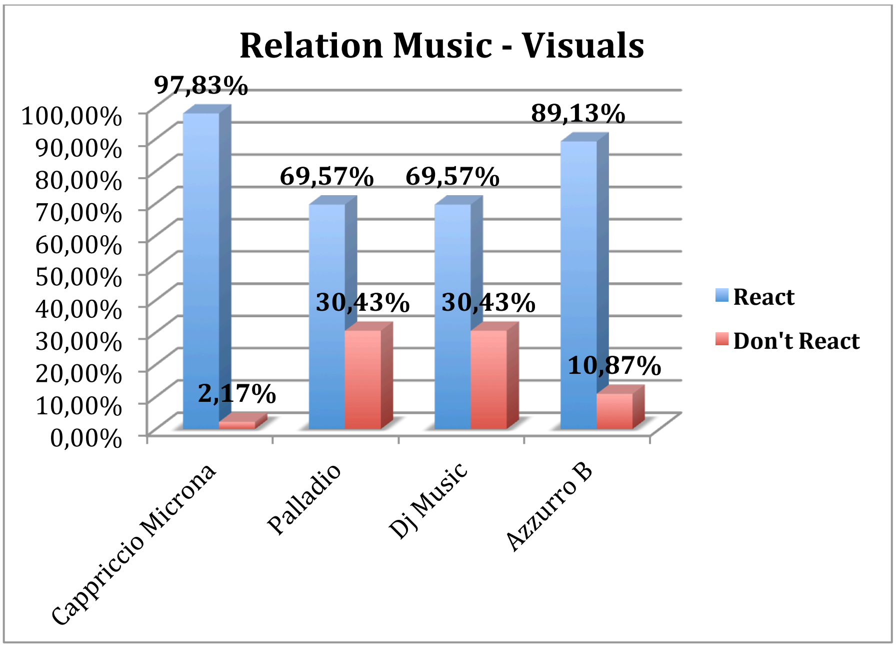


Figure 92 - Chart, which represents the answer to question 5 for people with music background

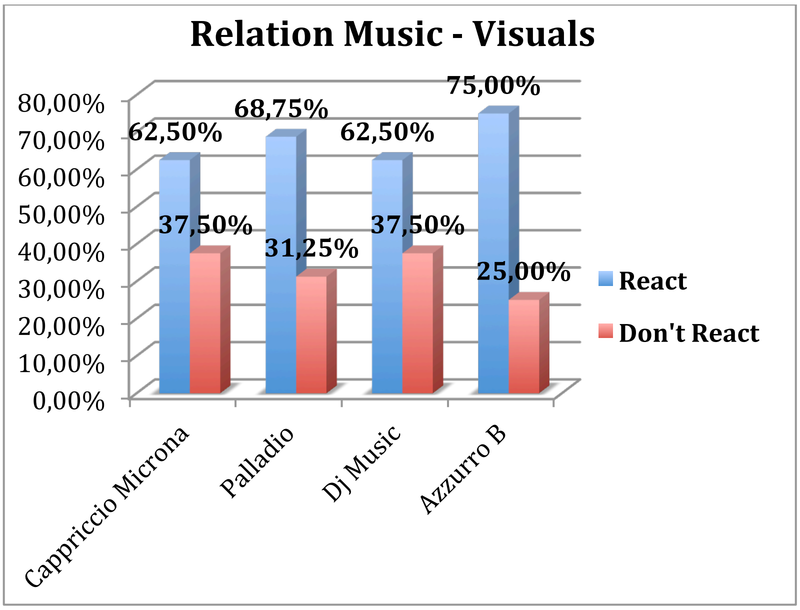


Figure 93 - Chart, which represents the answer to question 5 for people without music background

In figure 94 we can see that those people who thought that the music visualization from the Dj Scene was reacting with music, identified the duration element represented in the visualization (about 18 people). Additionally, this was the only music were the majority noticed duration in the visualization, in the other cases the element that people found was volume, specially in Cappricio Microna (about 34 people). The only music elements used for this project were volume and tonality. The other elements weren't used, so it is concluded that maybe people didn't understand well what they mean, or if they really can distinguish a visualization that is reacting with music or not. If the visualization changes over time (like the visualization for the Dj Scene), or the colors and changing, or even other parameters of the visuals are always changing. Then they may think that the music is triggering the changes in the visuals. The audience will try to imagine a connection between the music and the visualization even though there isn't any relationship. If the visual is always changing in a way that has always different results, then a part of the audience will not see the difference. This also could mean, that the visualization should be improved with better mapping between music and the visuals. With this survey, it is also possible to conclude that the volume is the best music element understood in the visuals.

In figure 95 we realize that the volume was again one of the major music elements detected by people (in Cappricio Microna 17 people noticed volume against 18 that noticed pitch). As for the DJ music, the main music elements detected were duration (16 people) and tonality (15 people). This is an example where people think that all visualizations are reacting with music, so even though the visual isn't reacting they try to find a connection. This music visualization wasn't reacting, so they imagined a connection between the visual and duration and tonality. If the audience thinks that the visuals are reacting with music, then they will observe a connection between music and the visualization. But we have to pay attention because not all people was mistaken and noticed that there wasn't a connection between the music and visualization.

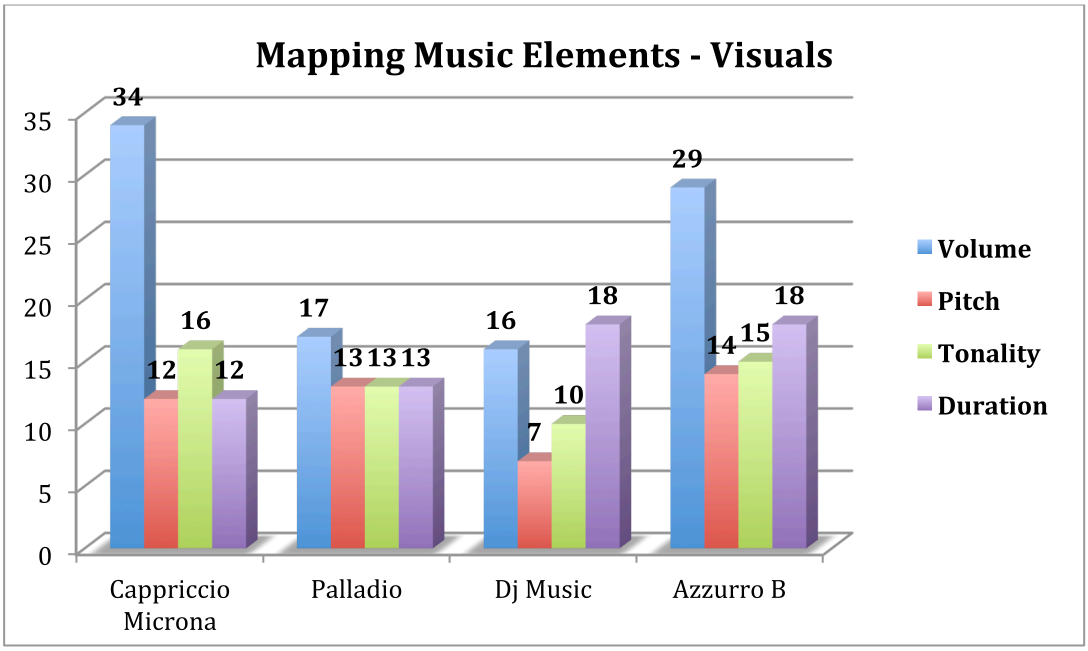


Figure 94 - Chart, which represents the answer for question 6 for people with music background

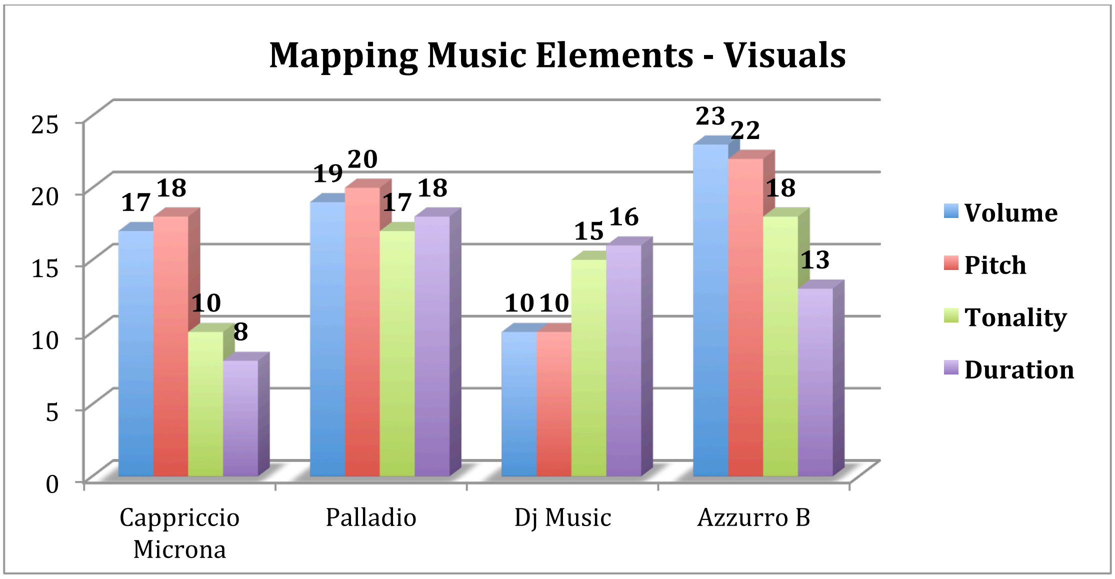


Figure 95 - Chart, which represents the answer for question 6 for people without music background

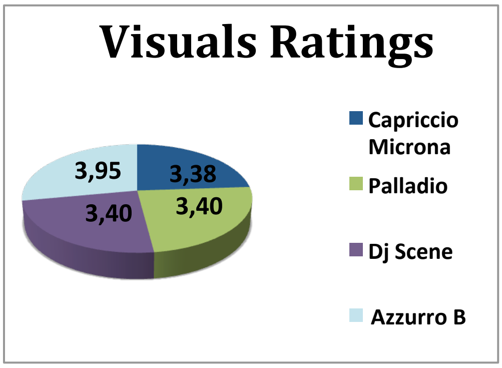
In short, the second goal was to prove the theory proposed previously "People are capable of noticing that the visuals are reacting with music". We conclude that, having a music background influences, because those people have a better understanding of music thus, they easily detect music elements embedded in the visualization such as: volume, pitch, tonality, and duration. When they realize changes in the volume, they quickly see it through the visuals. Whilst, people without music background don't have these music concepts well formulated, they might don't realize that some changes of the visuals are triggered by music elements. This is the only difference, between people with and without music background. Another interesting fact is that for both cases, almost the majority of people (superior to 68,75%), imagine relations between the music and visuals while they see a music visualization. If they know since the beginning that the visuals are reacting with music, then they will think that all visuals are reacting with music, even they are not.

Finally, it was decided to know the quality of the visuals. This is the reason why in the survey it was asked: "What do you think about this music visualization?" (Question 9 in Attachment A). We wanted to know if people liked the visuals, and what they thought about it. In order to do this, it was divided people with design experience from people without music experience, because people with design experience have better knowledge about design so they have a better opinion about the design of my visuals and also better advices or different design ideas, which can help me to improve the design in the future. And it was also decided to see what is the age range, which most appreciates this project. Consequently, we will know what is the target for this kind of project.

6.3. Third Goal

Again for the last goal, it was created a kettle file like the others. The difference is the output excel file where the chosen columns for the excel file are: the age, design experience (question 2), and the other visual they would imagine for that music piece (question 8).

We can conclude that the music animation with the best rating by people with design experience was the Azzurro B music piece with and average of 3,95 (see figure 96), and the music animation less rated was Cappricion Microna with an average of 3,38 (see figure 96). The scale for rating was 1 to 5 (1 very poor, and 5 very good). Thus, the quality of the visuals was average because the averages of ratings for all visuals are similar (3,38, 3,40, 3,95). On the other side, people without design experience (corresponds to 57% of the test users) rated the visuals like people with design experience. The visual most rated (like in the previous scenario) was Azzurro B with an average of 3,67 (see figure 96) and the less rated was Palladio with an average of 3,35 (see figure 96). In this scenario happened the same as the previous one, the average of the ratings from all the visuals are very similar (3,35, 3,50, 3,57, 3,67) as you can see in figure 96.



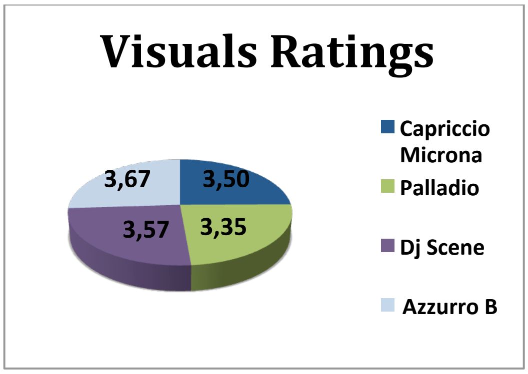


Figure 96 - Visuals Ratings of people with design experience (left) and without design experience (right)

Additionally, it was decided to know what is the age group of people, which most enjoyed the visuals. As we can see in figure 97, younger people (with design experience) are the ones who gave better ratings to the visuals. Also, people with more than thirty-six years old gave good rates for the visuals. In the other hand, people without design experience, gave higher rates to the visuals (see figure 97). All ratings were superior to three, and again the age group with the best ratings was again the younger ones (18 - 25).

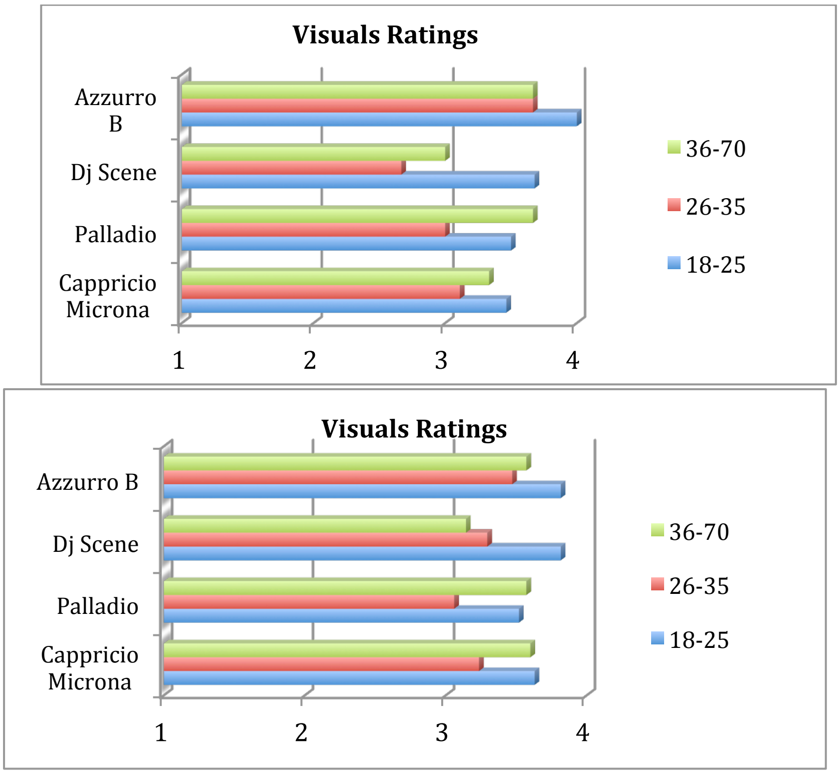


Figure 97 - Visual ratings for each age group of people with design experience (up) and without music experience (down)

7. Conclusion

With the development of this thesis was possible to achieve all intended goals, and also contribute for this area of music visualization. The principal goal of this thesis was to find the best mapping between music and visuals therefore, the connection between the audience and music in a live performance is enhanced. In order to achieve this goal, some theoretical hypotheses were developed and afterwards evaluated by an online survey.

In order to achieve the goal of discovering the best mapping functions was creating several mapping functions based on related work, and then test it in different live performances. Some of those mappings were altered, and the final result had the envisioned feedback from the spectators. This is the main contribution of this thesis, the best relations between music and visuals are: tonality – color; volume – position in y-axis; volume – size; volume – speed of movement. Attention, in the future new relations can be found and added to these relationships. All these relationships were proved through the several live performances in addition with an online survey, which helped knowing the feedback from people.

The online surveys facilitated the task of gathering the knowledge for validation of the hypotheses formulated in section 6. Summarizing, when somebody is creating a music visualization should know in detail the music piece and its story and feelings, because the majority of people when listening to that piece will feel the same. Therefore, the visuals have to be consistent with the music story, this is why it helps when the person who creates the music visualization also has a background in music. Because, people with music background have the ability to fully understand one certain music, moreover they can interpret music in a different way. We also know, that personal life experiences influences our feelings, and sometimes we can relate a special music piece to a determinate moment of our lives. This is the reason why the audience sometimes has another kind of feelings when listening music. They relate it to a moment occurred previously in their lives. All of these factors have to be considered before starting a music visualization composition in order to design visuals in accordance with the music. Otherwise, if this step is not created correctly, then this all experience will not be significant, as there will not exist relationship between music and visuals, thus audience will not be able to connect to the live performance, and the visuals will become a distraction.

After this phase, it was decided to know if people are able to notice that the visuals are reacting with music. With the survey was possible to conclude that if people already know the project, and know that the visuals are reacting to music, then they will try always to find a connection between music and visuals. Even though there is not a relationship, they try to imagine one. Some people can detect that the visuals aren't reacting, but the majority thinks the visual are reacting, despite the fact if they have experience with music or not. Anyway, even if the visuals are reacting or not, they should help the audience comprehend the music involving them with music feelings. This is something to work in the future, visualization should be an extension of the music and should help people really connect with the live performance, but makes difference if they are reacting to music or not? The answer for this question, is that it doesn't matter if they are reacting or not, the intentions is to convey the music feelings, and should be synchronized with the music in real-time, nevertheless if the graphic animation is reacting to music it will make this process easier, because with music visualization the process is automatic. In other words, if there is a clear mapping between sound and visuals, then we'll have high responsiveness. Interesting visuals that keep developing and changing over the course of the music piece, this means that the visual will remain interesting by continuously introducing novelty to it.

The detected visuals that were reacting with music, it was identified volume in almost every visuals, and also pitch and duration. This is something really difficult to know, because people outside the music world are not familiar with these music terms, such as pitch, or tonality. And these are music elements, which normally they don't have an idea of what this is. Regular people only have knowledge of volume (or dynamics) and pitch, this is why they might have difficulties identifying those music elements in the visuals. Nevertheless, in the online survey there was an example for each music element with the intention of teaching people what those elements are, and if they can detect them through the visuals. As people don't now they are, they will not find them in the visuals, this is why the volume is the main element that has to be present in the visuals. Even they don't know what is tonality, this element should be present because helps identifying the mood of music, thus should be mapped into color in order to give the appropriate atmosphere related to the music mood.

So, there are some music elements that people might not know what they are, but they should be present because they help with the connection improvement between the spectators and the live performance with music visualization, and tonality should be one of those elements to be present. This is one possible way of detecting different episodes within music, but is not enough, this is why it was decided to have a person triggering the animations in the different music episodes by pressing a key from the keyboard.

Another interesting discovered fact is that younger people (18-25) is the age group who most found this kind of technology fascinating because they were the age group who gave better ratings to the visuals. This is good, because it is a good alternative to bring more youngsters to any kind of concerts. Usually, younger people don't appreciate classical music concerts, or even people with more advanced ages. But with music visualization, the experience of a classical music concert or any kind of life performance is different, more intensive. Music visualizations awakes new feelings in the audience, they found it more interesting as they have a different connection with music. And this is the reason why some people came to these performances even though they don't like this kind of music.

With the online survey, it is possible to conclude with the test subjects comments' that music visualization is an extension to the live performance. They say that through visual means is easier to identify with the music piece and with what it is pretended to convey. The visualizations have the capacity of engaging the audience feelings with the live music feelings, and this makes the live performance richer and fascinating. In other words, the visuals adds something new to the performance, it is like a complement for the concert, enabling an improved experience for those who goes to a live music performance. Additionally, music visualization give a better understanding of music to the spectators, even those who haven't got any kind of music knowledge, is easier for them to follow music in a live performance. Furthermore, this is good for those people who can't hear, because with the visualization those people can visualize and interpret what is going on and create their own mental model of music.

Still, there is some work to do in this theme. Despite the fact that the suitable mapping functions were created in this thesis and proved with the test subjects of the online survey, other kinds of mappings can be introduced. These are the main basis mapping functions for creating music visualization. The pitch detector used for this thesis is not very accurate, so it should be improved as well as the visuals. The visuals should be always improved, there is always something to adjust or some detail to change. Also, other kinds of graphical animations could be used, as the test subjects of the survey mentioned that they would have a greater connection with music if it were used real images instead of abstract animations. This is something to think about, because some people's feedback was that with real images they were able to connect to music. So instead of using abstract visuals as a universal music visualizer, it should be used specific images according to that music. The objective was to be as flexible as possible, but maybe using real photos people can create a more intensive level of engagement with the live music performance. Those photos can also react to music, but in that case new mapping between photos and image should be created, for example, perhaps the color of the picture can continue to be related to tonality, and the size with the dynamics.

Another interesting research for future work in music visualization, should be finding new kinds of interaction, always with the goal of enhancing the live music performance. For instance, use other technologies such Kinect[7] to detect movements and then trigger the graphic visuals associated. Imagine having a Kinect to detect the movements from the conductor, as you know, the conductor movements are meaningful for the musicians, because he expresses what he wants in a certain moment for a music piece. Normally, those movements are the dynamics, which the musicians should be playing, so this is another way of capturing music elements through the conductor's movements.

Moreover, it should be interesting joining brain computer interface with music visualization. Brain computer interface is a technology, which establish a communication between the brain and external devices through a computer. With this kind of technology, it is possible to control systems using "thoughts". For example, with this, is possible to detect what kind of feelings the user has at the moment, and with this knowledge, it possible to identify the mood of music by what the conductor or a musician is feeling when he conducts or plays that music piece. Which then would trigger different kinds of visuals and would create a different level on intensity between the music and the musicians, resulting different kinds of visuals responding to the musician feelings captured through the brain computer interface. These are different methods to perceive music elements and trigger different kinds of visuals, which might improve the connection between the audience and the visuals generated by the music perception in real-time of musicians.

[7] http://www.xbox.com/en-US/KINECT

Finally, the next step is having different visuals for different instruments, which has as drawback the kind of music, which is being played. Imagine, that one instrument only plays at the beginning of the music piece, or that he has the same rhythm the whole piece (sometimes happens with bass instruments). The visual will not have the necessary changes to make the visualization interesting. This is why, visuals should be chosen considering the music pieces and all its changes throughout the piece, in order to provoke different visuals.

Bibliography

Adobe Systems Incorporated. (2012). *Adobe*. Retrieved January 14, 2012, from Adobe After Effects CS5.5: http://www.adobe.com/products/aftereffects.html

Apple Inc. (2012). *Developer Apple*. Retrieved September 20, 2011, from Graphics and Animation: http://developer.apple.com/technologies/mac/graphics-and-animation.html

Apple Inc. (2007, July 17). *Mac OS X Developer Library*. Retrieved September 20, 2011, from Quartz Composer User Guide: http://developer.apple.com/library/mac/#documentation/GraphicsImaging/Conceptual/QuartzComposerUserGuide/qc_concepts/qc_concepts.html#//apple_ref/doc/uid/TP40005381-CH212-DontLinkElementID_1

Apple Inc. (2007, July 07). *Mac OS X Developer Library*. Retrieved September 20, 2011, from Quarz Composer User Guide: http://developer.apple.com/library/mac/#documentation/GraphicsImaging/Conceptual/QuartzComposerUserGuide/qc_concepts/qc_concepts.html#//apple_ref/doc/uid/TP40005381-CH212-DontLinkElementID_1

Apple Inc. (2007, July 07). *Mac OS X Developer Library*. Retrieved September 20, 2011, from Quartz Composer User Guide: http://developer.apple.com/library/mac/#documentation/GraphicsImaging/Conceptual/QuartzComposerUserGuide/qc_tips_moves/qc_tips_moves.html#//apple_ref/doc/uid/TP40005381-CH203-TPXREF101

Apple Inc. (2008, October 15). *Mas OS X Developer Library*. Retrieved September 22, 2011, from Quartz Composer Programming Guide: http://developer.apple.com/library/mac/#documentation/GraphicsImaging/Conceptual/QuartzComposer/qc_intro/qc_intro.html#//apple_ref/doc/uid/TP40001357

Bain, M. N. (2008). *Real Time Music Visualization: A Study in the Visual Extension of Music.* The Ohio State University, Department of Industrial, Interior, and Visual Communication Design. Ohio State University and OhioLINK.

Cycling '74. (2012). *Cycling '74*. Retrieved September 20, 2011, from What is MAX?: http://cycling74.com/whatismax/

Cycling '74. (2012). *Cycling '74 creates software for innovation and exploration*. Retrieved 2012, from Company ‹‹ Cycling 74: http://cycling74.com/company/

DownloadAtoZ. (2011). *VDMX 7.44b for mac*. Retrieved October 17, 2011, from Mac.DownloadAtoZ: http://www.google.pt/imgres?q=VDMX&um=1&hl=en&sa=N&biw=1199&bih=635&tbm=isch&tbnid=D51fUvA5wrZjQM:&imgrefurl=http://mac.downloadatoz.com/vdmx/&docid=asPnLQxcsMsweM&imgurl=http://mac.downloadatoz.com/resources/soft/mac/download/img/v/d/vdmx-demo.jpg&w=600&h=375&ei=z44iT4yqDtKv8QPhhenSBw&zoom=1&iact=rc&dur=292&sig=101262971091596753895&page=1&tbnh=136&tbnw=193&start=0&ndsp=15&ved=1t:429,r:0,s:0&tx=64&ty=130

Kosada Incorporated. (2011). *Kineme | Quartz Composer Stuff*. Retrieved from Kineme Interactive Media: http://kineme.net/

Lillie, A. (2008). *flyingpudding*. Retrieved January 07, 2012, from anita lillie : flyingpudding.com : Visualizing Music: http://flyingpudding.com/projects/viz_music/

Nullsoft. (2012). *Winamp*. Retrieved 2012, from Winamp Media Player - MP3, Video, and Music Player - Winamp: http://www.winamp.com/

Pure Data. (2012). *Fourier Analysis* . Retrieved 2012, from Pure Data: http://www.pd-tutorial.com/english/ch03s08.html

Pure Data. (2012). *pd~*. Retrieved 2012, from Pure Data - PD Community Site: http://puredata.info/

Resolume. (2012). *Resolume*. Retrieved December 2, 2011, from Resolume VJ Software - Live Digital Motion Graphics: http://www.resolume.com/

The Echo Nest Corporation. (2012). *The Echo Nest*. Retrieved January 07, 2012, from The Intelligent Music Application Platform - The Echo Nest: http://the.echonest.com/

vBulletin Solutions Inc. (2011, May). *Resolume Avenue v3.3.2*. Retrieved October 17, 2011, from Techfuels Technology savy fuel: http://www.google.pt/imgres?q=resolume+avenue&um=1&hl=en&sa=N&biw=1199&bih=635&tbm=isch&tbnid=DwHz2UIuFCur3M:&imgrefurl=http://www.techfuels.com/video-card/65628-resolume-avenue-v3-3-2-portable.html&docid=hkDt0S0PST8OoM&imgurl=http://www.techfuels.com/attachments/video-card/27126d1321857885-resolume-avenue-v3-3-2-portable-resolume-avenue-v3.3.2-portable.jpg&w=500&h=312&ei=5pAiT4mGHoPL8QPFypi7CQ&zoom=1&iact=hc&vpx=300&vpy=157&dur=735&hovh=177&hovw=284&tx=124&ty=125&sig=101262971091596753895&page=1&tbnh=133&tbnw=193&start=0&ndsp=16&ved=1t:429,r:1,s:0

VideoLan Organization. (2012). *VLC media player*. Retrieved 2012, from VideoLAN - VLC: Official site - Free multimedia solution for all OS!: http://www.videolan.org/index.html

VIDVOX LLC. (2012). *VIDVOX*. Retrieved October 2, 2011, from VDMX5 VIDVOX: http://www.vidvox.net

Vimeo. (2012). *Vimeo*. Retrieved December 2, 2011, from Quartz Composer/VDMX tutorial no 1: The bouncing ball: http://vimeo.com/2100787

Windows. (2012). *Windows Media Player - Microsoft Windows*. Retrieved 2012, from Windows Media Player: http://windows.microsoft.com/en-US/windows/products/windows-media-player

Wonder How To, Inc. (2011). *WonderHowTo*. Retrieved October 2, 2011, from How to use Quartz Composer together with Resolume Avenue on your Mac: http://www.wonderhowto.com/how-to-use-quartz-composer-together-with-resolume-avenue-your-mac-416459

E. Isaacson, "What you see is what you get : On Visualizing Music," in *Cumulative ISMIR Proceedings*, 2005.

B. Klemenc, P. Ciuha, and L. Subelj, “Visual and Aural : Visualization of Harmony in Music with Colour,” *Transactions on Internet Research*, vol. 7, no. 1, 2011.

R. Hiraga, R. Mizaki, and I. Fujishiro, “Performance Visualization – A New Challenge to Music Through Visualization,” in *MULTIMEDIA ’02 Proceedings of the tenth ACM international conference on Multimedia*, 2002.

R. Taylor, P. Boulanger, and D. Torres, “Using music to interact with a virtual character,” in *NIME ’05 Proceedings of the 2005 conference on New interfaces for musical expression*, 2005.

D. Cooke, *The language of music*. Oxford University Press., 1963.

O. Kubelka, “Interactive music visualization,” in *Central European Seminar on Computer Graphics 2000*, 2000.

C.-han Chen, M.-fang Weng, S.-kang Jeng, and Y.-yu Chuang, “Emotion-based Music Visualization using Photos,” in *MMM’08 Proceedings of the 14th international conference on Advances in multimedia modeling*, 2008.

H.-H. Wu and S.-kang Jeng, “Probabilistic estimation of a novel music emotion model,” in *he 14th International Multimedia Modeling Conference*, 2008.

E. Chew and A. R. J. Franc, “MuSA . RT : Music on the Spiral Array . Real-Time,” in *MULTIMEDIA ’03 Proceedings of the eleventh ACM international conference on Multimedia*, 2003.

P. Ciuha, B. Klemenc, and F. Solina, “Visualization of Concurrent Tones in Music with Colours,” in *MM ’10 Proceedings of the international conference on Multimedia*, 2010.

N. Herman and N. I. Badler, “Movies from music: Visualizing musical compositions,” in *SIGGRAPH ’79 Proceedings of the 6th annual conference on Computer graphics and interactive techniques*, 1979.

R. Hiraga and N. Matsuda, “Visualization of Music Performance as an Aid to Listener’s Comprehension,” in *AVI ’04 Proceedings of the working conference on Advanced visual interfaces*, 2004.

S. C. Nanayakkara, E. Taylor, L. Wyse, and S. H. Ong, “Towards Building an Experiential Music Visualizer,” in *Information, Communications & Signal Processing, 2007 6th International Conference on*, 2007.

S. M. Smith, “A Visualization of Music,” in *Visualization ’97*, 1997.

P. D. Cuadra, A. Master, and C. Sapp, “Efficient Pitch Detection Techniques for Interactive Music,” in *International Computer Music Conference*, 2001.

Attachments

Survey Questions:

What is your age?

1. Do you have any music experience? (Play an instrument, teach music classes…)

2. Do you have any design experience? (Photography, WebDesigner, Creating: Posters, Flyers, Animations)

Listen Audio 1 and answer the following questions.

3. What do you feel listening this music?

4. What do you imagine when listening to this music?

Watch Video 1 and answer the following questions.

5. Did you notice relations between this music piece and the visualization?

6. If yes, what music elements can you observe through these visuals?

7. Would you imagine different visuals for this kind of music?

8. If yes, what kind of visuals do you imagine for this music piece?

9. What do you think about this music visualization?

Watch video 5 and based on other concerts with my music visualizations you perhaps saw, answer the following questions.

10. In your opinion, what makes interesting going to a live performance with music visualization?

11. A live performance with music visualization influences your decision of going to that performance?

12. What is your opinion about the visuals?

13. Do you have any suggestions or recommendations?