# Hexaphonic Guitar Transcription and Visualization

**Iñigo Angulo Otegui**

MASTER THESIS UPF / 2016

Master in Sound and Music Computing

Master thesis supervisors:

Rafael Ramirez

Sergio Giraldo

Department of Information and Communication Technologies

Universitat Pompeu Fabra, Barcelona

UNIVERSITAT POMPEU FABRA

# Acknowledgements

I would like to thank my supervisors, Rafael and Sergio, for taking on my project and for their advice during it´s elaboration. Also, many thanks to all the people who were part of this life stage. Lastly, to my family, for giving me the chance and their unconditional support.

# Abstract

Music representation has been a widely researched topic through centuries. Transcription of music through the conventional notation system has dominated the field, for the best part of the last centuries. However, this notational system often falls short of communicating the essence of music to the masses, especially to the people with no music training. Advances in signal processing and computer science over the last few decades have bridged this gap to an extent, but conveying the meaning of music remains a challenging research field. Music visualization is one such bridge, which we explore in this work. This research presents an approach to visualize guitar performances, transcribing musical events into visual forms. To achieve this, hexaphonic guitar processing is carried out (i.e. processing each of the six strings as an independent monophonic sound source) to get music descriptors, which reflect the most relevant features of a sound to characterise it. Once this information is obtained, our goal is to analyse how different mappings to the visual domain can meaningfully/intuitively represent music. As a final result, a system is proposed to enrich the musical listening experience, by extending the perceived auditory sensations to include visual stimuli.

# TABLE OF CONTENTS

# FIGURES

# TABLES

# 1 INTRODUCTION

Music is one of the most powerful art-expressions. Through history, humans have shared the musical realm as part of their culture, with different instruments and compositional approaches. Often, music can express what words and images cannot and thus remains a vital part of our daily life. Advancements in technology over the last decades have brought us the opportunity to go deeper into developing an understanding of music, in the context of other senses such as sight, which dominates over other senses for representing information. In this work we propose a system to extend music by developing a visualization or visual notation approach to map the most important features that characterise musical events into the visual domain.

The idea of providing mechanisms for understanding music using our eyes is not new, as traditional music notation (i.e. scores) may provide us with an idea about the acoustic content of a piece without the need of previously listening to it. However, this project´s approach is not intended to design a performance instructor, but a visual extension of the musical events that compose a performance/piece. The goal is to develop a system that is able to visually represent the musical features that best characterise the music produced by a guitar, that is, to develop a real-time visual representation system for guitar performances.

One challenge for the development of such a system is the polyphonic nature of the guitar. The complexity of polyphonic sound transcription is well known, so to solve this issue, the use of a hexaphonic guitar is chosen, in which each of the strings is processed as an independent monophonic sound source, simplifying the transcription of the sounds. Two methods of transforming a conventional guitar into a hexaphonic one are proposed. Once the desired musical features are obtained, different ways to represent them are studied, analysing the mappings between sound and visual dimensions.

The aim of this work is to offer a tool in which information about the musical events (e.g. pitch, loudness, harmony) of a guitar performance is visualized in real-time through a graphical user interface (GUI). The visualization of this information is approached from two different perspectives. Firstly, trying to accurately reflect the guitar performance events in the GUI, providing objective information about them (such as the notes being played, harmony or chords derived from those notes, etc), allowing the user to learn accurate information from it. And secondly, setting impressive/artistic mappings between sound and visual domains to create an aesthetic experience for the user. This kind of visualizations could enrich music experience by accompanying musical events with visual stimuli.

## 1.1 The main tasks of the project

This project is divided into three main tasks.

- The first one relates to the hardware construction of the hexaphonic guitar, for which two different approaches are proposed in order to transform a conventional guitar (both classical and electric guitar) to achieve hexaphony.

- The second task consists of guitar music transcription. The techniques and software tools to track the most relevant musical features of guitar music are described.

- Lastly, in the third part, the data extracted from the transcription task is used to create the visualizations. Different designs are proposed for the visualizations to explore different applications.

The final result of the project consists of a system to include all the tasks. A system able to take the audio signals from the hexaphonic guitar, process them to obtain information about the musical events that occur in it, and visualize a representation of this information on a user graphical interface (GUI), based on different music-to-visuals mapping hypothesis.

## 1.2 Research Question

Summarizing the aforementioned ideas, the research question posed in this project could be *how possible is to obtain a visual representation of the music played with a guitar, and how could different visualization approaches enrich the music experience of the listener/viewer*.

# 2 STATE OF THE ART

This section is divided according to the project task classification previously presented. Firstly, some background about music transcription concepts and methods is presented, and examples of systems dealing with guitar music are analysed. Then, music visualization is introduced by giving some historical account, explaining relevant concepts to the topic and giving some examples of systems for visualizing music.

## 2.1 Transcription

Transcription of music is defined as *"the process of analyzing an acoustic musical signal so as to write down the musical parameters of the sounds that occur in it*" (Klapuri, 2004).

The goal of transcription is to represent music as detailed as possible, so that it can be accurately reproduced. Through the last centuries, scores have become the most widespread transcription system. These use written symbols to describe the sounds that each of the instruments within a piece produces. Thus, the played notes along time are represented, detailing their pitch height, onsets, and durations. Loudness information is not usually specified for individual notes, but for larger parts of the piece.

However, Klapuri (2004) defends that the applied notation in music transcription has not necessarily to be traditional one, but any symbolic notation that transmits the sufficient musical information for interpreting the piece. Nowadays, there exist many alternative approaches to music transcription. As an example, one could think about MIDI piano-roll representations, which could obtain the best results not only in terms of accuracy and precision, but also flexibility and easiness when modifying the piece.

When talking about musical information, we deal with a very wide concept, as numerous point-of-views are equally valid to describe and understand it. As Argenti, Nesi & Pantaleo (2011) stated, musical information is very multi-faceted, since it includes different levels of information. Different levels of musical information correspond to the different levels in which humans can perceive and understand music. Low-level information corresponds to the physical parameters we use to measure sound. These physical parameters find their equivalents at the human sensorial level, built from the interpretation our brain performs from sound. Mid-level musical information is derived from the sensorial level, comprising the human perceptual interpretation of music from the sensorial stimuli. Finally, high-level information corresponds to the structure we apply for understanding and arranging music, which comes from previous knowledge acquired with experience. At this level we can also find the emotions evocated from music expression. Figure 1 reflects the different levels for musical information.

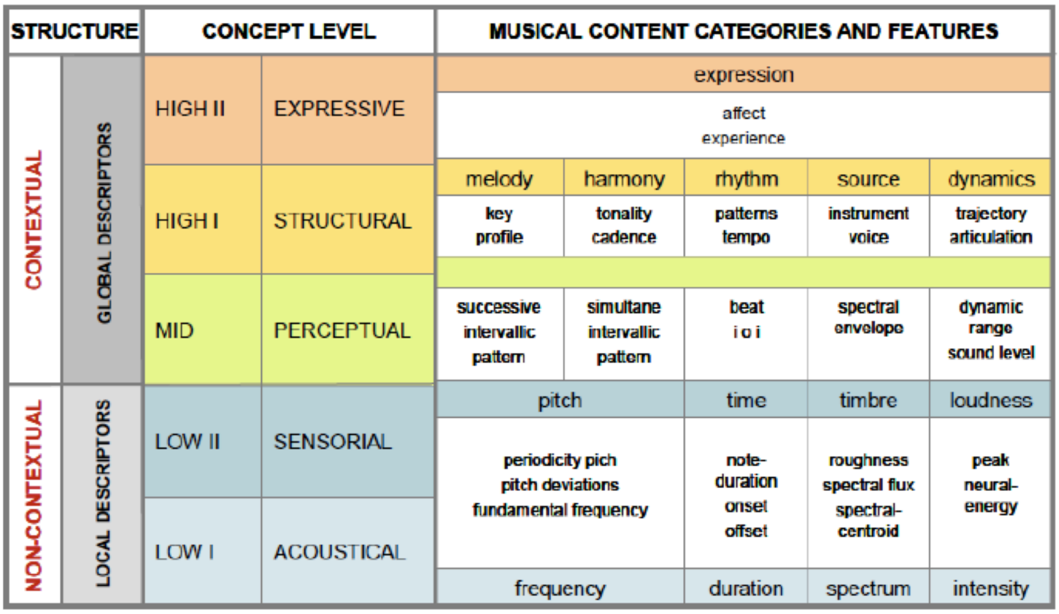



**Figure 1. Music information descriptors classified by levels (Lessafre, 2005).**

Music transcription approaches could be difficult to categorize, as human comprehension of music is understood to use these three information levels, and more specifically a combination of them. Several methods, as the bottom-up/top-down processing, or the differentiation at signal analysis level (time domain analysis vs. frequency domain), are described by the aforementioned authors (Argenti, Nesi, & Pantaleo, 2011).

However, the major part of current music transcription systems share some characteristics and approaches to fulfil their mission. A general overview is presented in the next section.

## a) Automatic Music Transcription System

An Automatic Music Transcription System is that one capable of performing the transcription task explained in the previous section: given an input audio signal, produce a notation reflecting the most relevant information about the musical events within it, as an output. Figure 2 reflects a block diagram with the general architecture of a music transcription system.

As we can see in the diagram, the input of the system is an audio signal, which is pre-processed using different techniques. Usually, segmentation is applied here to obtain a frame divided representation of the signal, as well as other techniques to compute mid-level representations, such as spectral analysis, auditory model based representation, etc. Then, different information about musical notes is computed by dedicated blocks, often separated by pitch and time information estimation. Additional knowledge sources could be used to achieve better transcription performances, such as harmonic and/or instrumental models, training databases, etc. Finally, all the extracted information is post-processed in order to structure it in the expected output representation format (such as MIDI piano roll, traditional score…).

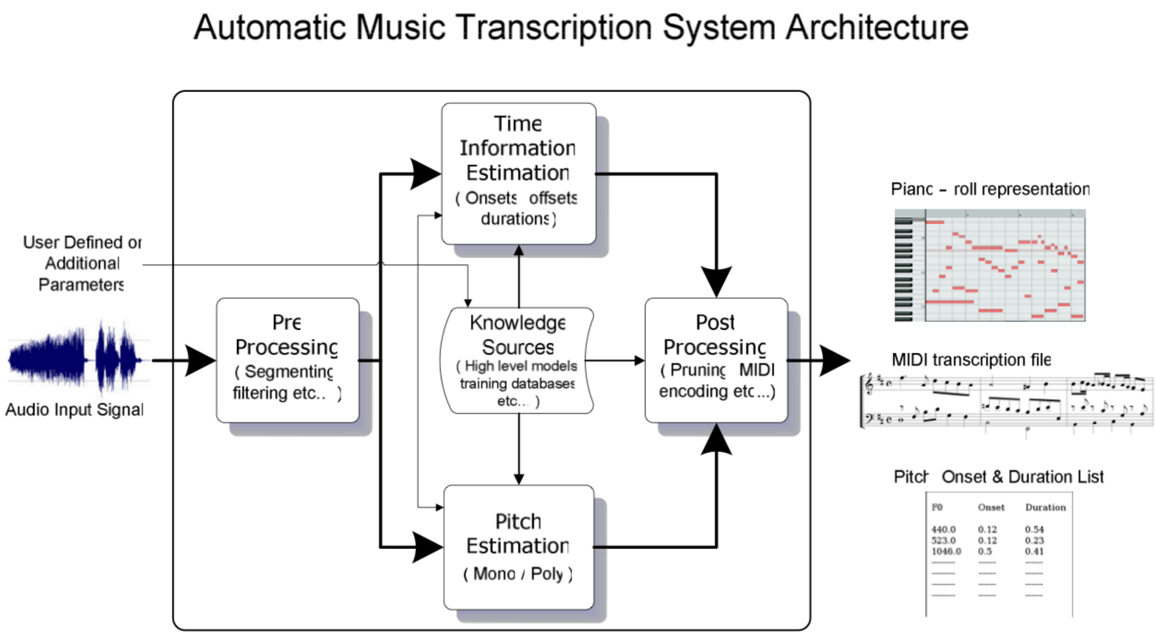



**Figure 2. General architecture of an automatic music transcription system (Argenti, Nesi & Pantaleo, 2011).**

A great variety of automatic music transcription systems have been proposed over the years, involving different approaches and techniques. A more detailed analysis and comparison between their performance is offered by Argenti, Nesi & Pantaleo (2011).

## b) Monophonic vs. Polyphonic Sources

When working within music transcription field, a major distinctive cue of the transcription problem is given by the number of voices a music piece have, or from the point of view of the analysed signal, the number of sound sources that are present in it. Having a single sound source in our signal, or at the contrary multiple sources, means facing two very different transcription problems, in terms of the strategy to apply to solve them and the complexity of this strategy. This project focuses on a single instrument, but it is essential to understand if the instrument we are dealing with supposes a monophonic or a polyphonic sound source.

A monophonic sound source is that one capable of producing one single sound at a time. In contrary, a polyphonic source can produce multiple sounds at the same time. There exist both monophonic and polyphonic musical instruments. An example of a monophonic instrument could be the flute. A single flute player cannot produce two different notes simultaneously. So, from a single monophonic instrument, in this case the flute, only one musical note can be heard at a time. On the other hand, examples of polyphonic instruments could be the piano or the guitar. In the case of the piano, for example, the polyphony is due to two or more keys pressed at the same time, each of them producing a different note in an independent string. So, from a single polyphonic source, we can hear two or more notes played simultaneously.

Monophonic and polyphonic source transcriptions use different techniques, and generally, the first one obtains better results in terms of accuracy, due to the complexity of polyphony analysis. In fact, in the former case pitch tracking is practically considered to be a solved problem within the state-of-the-art techniques (Klapuri, 2004). However in the case of polyphonic source transcription, it is further from being successfully settled, and additional difficulties arise in presence of multi-instrumental contexts. Difficulties arise in polyphonic music transcription since two or more concurrent sounds may contain partials that share the same frequency values. Problems like overlapping may appear, and lead to the use of other signal processing techniques, often combined with a priori knowledge resources (additional knowledge sources shown in Figure 2 block diagram) (Argenti, Nesi, & Pantaleo 2011).

## c) Guitar Transcription

Guitar transcription is the process of creating a human-interpretable musical notation from the sound signal it produces. When dealing with guitar music, the transcription process is particularly complex due to the polyphonic nature of the sound it emits. This polyphony is caused by the different strings of the guitar played together, which leads to several notes sounding at the same time (chords).

For the pitch estimation part of the transcription, detecting multiple fundamental frequencies is a difficult task. The techniques used for monophonic transcription, which seem robust and have pretty accurate results, become very complex to apply for polyphony, or even impossible, leading to search for different techniques to overcome the problem. For example, when analysing the spectrum of a guitar sound, "*it may not be clear whether a peak in frequency is a fundamental or a harmonic, or both*" (Fiss & Kwasinski, 2011).

### Pitch Tracking Techniques for Monophonic Guitar

There exist several techniques to approach the pitch estimation task for monophonic sources. These are normally divided into two main groups: time domain and frequency domain techniques. Time domain techniques deal with the representation of the signal over time and are normally based on measuring periods in the wave, which are inversely correlated with the frequencies that compose the signal. On the contrary, frequency domain techniques normally deal with the spectrum of the signal. The spectrum results from applying the Fourier Transform to the time domain signal, which leads to the frequency domain representation of it, reflecting the frequencies that compose it.

Within the state-of-the-art techniques for the pitch tracking of monophonic sources, there exist several approaches, such as the autocorrelation method, Yin and spectral peak picking, among others (Knesebeck & Zölzer, 2010).

**Polyphonic guitar transcription**

Nonetheless, there also exist approaches for transcribing polyphonic guitar music. One example could be the system proposed by Fiss & Kwasinski (2011), whose aim is to perform an automatic electric guitar audio transcription in real-time. In this approach, the authors use the STFT (Short Time Fourier Transform) to compute the spectrogram of the signal and extract information about the peak locations, which correspond to the frequencies of the waves that compose the signal (fundamental frequencies and harmonics). Then they compute the produced notes taking into an account the different possibilities of producing them among the six strings of the guitar, and thus, avoiding the ambiguity the guitar polyphony adds to the transcription.

However, due to the complexity of polyphonic transcription methods, for this project we opted to use monophonic audio transcription. To achieve this with a guitar, a hexaphonic guitar is needed. This approach keeps the transcription task simpler and leaves more time to study the visualization part.

## d) Hexaphonic Guitar

Conventional guitars usually have six strings. These strings are plucked using a pick or the fingers producing a vibration along them. In guitars that have a line output, i.e. electric and electro-acoustic guitars, the vibration of the strings are captured by the pickup electronics. Common guitar pickups sum all the strings´ vibrations into a single signal, which forms the output line signal of the instrument. This leads to a signal composed of multiple voices, corresponding to the different strings, and thus producing polyphony.

Nowadays, another approach is attracting more and more attention, i.e. hexaphonic guitar. A hexaphonic guitar consists of six monophonic sound sources, instead of being seen as a polyphonic instrument. This means that the sound of each string is separated and independently outputted from the instrument.

This kind of guitars offers many possibilities in comparison with traditional electric guitars. Each string can be processed independently, from the point of view of the final guitar audio mix signal. Different audio effects can be applied to particular strings, or simply volume and panorama variations may permit to create limitless sound images, allowing the musician to experiment with new textures and type of sounds.

And of course, from the point of view of music transcription, things seem to become simpler since polyphony is avoided. As monophonic source transcription is considered a much accessible problem in terms of complexity, this transcription approach for guitars could suppose a way of ensuring the performance of the system.

**Hexaphonic Guitar Transcription**

Related work to music transcription using hexaphonic guitars already exist. In this section an overall vision of a hexaphonic guitar transcription system by O'Grady & Rickard (2009) is presented.

In this work, the authors adapted a standard electric guitar to achieve hexaphony by using the Roland GK-3 divided pickup, and building a Breakout Box circuit to separate the sound of each of the strings. For the transcription task, they used Non-Negative Matrix Factorization (NMF), which is a method to decompose multivariate data, where a non-negative matrix, V, is approximated as a product of two non-negative matrices, $V \approx WH$.

With this, they built a basis matrix W by applying NMF to the training data and learning a fixed basis $W_1, \ldots W_6$ , for each string, where each column contains a magnitude spectrum that corresponds to the notes of that string. The training data consisted on recordings, for the different strings, that contained the notes present on each of them (all the notes that can be played on a particular string of a guitar, starting from the open string and then progressing all the way up and down along the fretboard playing all the notes). Then the recordings to be transcribed $V_1, \ldots ,V_6$ , were fitted to $W_1, \ldots ,W_6$ , resulting in activation matrices $H_1, ... , H_6$ , which indicated the position in time each note was played.

From these matrices a piano roll representation was build as a result of the transcription, and later transformed to a MIDI file to listen to the transcription result (O'Grady & Rickard, 2009).

## 2.2 Visualization

Visualization could be understood as the communication of data, which originally is presented as abstract or at least not immediately visible. Normally, visualization produces an image, but is not always clear, as in some arts the means of communication are based on other aspects, as for example the performer´s body expression in dance. However, within the focus of this project, we assume that visualization consists of imagery generated from the data to be communicated in a readable recognizable way. This data is extracted from musical features, so the visualizations reflect the evolution of music over time.

Throughout history, many artists, composers, scientists and inventors have been challenged by the relationship between visual and music art forms. This connection between art modalities has lead to a large number of different works and artistic expressions, such as painters who tried to paint music or musicians who did the opposite by trying to produce visuals from music. Often, this kind of works has been related to synaesthesia.

Synaesthesia is a neurological condition in which stimulation of one sensory or cognitive pathway leads to automatic, involuntary experiences in a second sensory or cognitive pathway. According to the definition, people with this condition can experience certain stimuli across their senses. Some of the artists that first created this kind of multimodal works were considered synesthetic. A famous example was Wassily Kandinsky, who, fascinated with the emotional power of music, set out to try and recreate it in painting. He used musical names to describe his works, and dealt with musical concepts, such as harmony, rhythm, dissonance, etc. in abstract painting (Bergstrom, 2011).

According to synaesthesia theory, sense perception can be decomposed into discrete units, whereby one sensation´s characteristic features find their equivalence in another´s. Music has its notes and phrases, harmony, and compositional structures; these features could find similarities with visual form, colour, space, and motion (Bain, 2008).

The interest in music visualization is to extend the experience of music hearing to sight. The idea of fusing music and visuals leads to the creation of a synesthetic experience for the listener/viewer, which may induce a stronger sensation due to multimodal perception over senses.

## a) Historical Account on Music Visualization

Throughout history, many machines have been created to explore the relationship between music and visual fields. In this section some of the first and most famous works are described.

One of the first inventions that merged sound and visual expressions was the “colour organ” (also known as “light organ”). Louis Bertrand Castle created the first light organ in 1730´s, which consisted of a clavichord with a screen placed above and small windows with coloured glass. The windows were covered with curtains, which avoided a light placed behind to pass through them. Each key of the instrument had a curtain attached and, when one was pressed, it allowed the light to shine through the glass (Bain, 2008).

Another well-known example was the creation of the Clavilux that consisted on a similar invention to the colour organ, which produced light projections for music. The Clavilux was created in 1921 by Thomas Wilfred, and the art form was called “Lumia” (Bain, 2008). However, the peculiarity of this system was that, unlike most of his predecessors, rejected the notion that there is any absolute correspondence between music and visual art, and instead concentrated only on light. Wilfred separated the elements to describe a Lumia performance into colour, form and motion, in contrast to the earliest pioneers of the art form, who only considered colour in their compositions (Bergstrom, 2011).

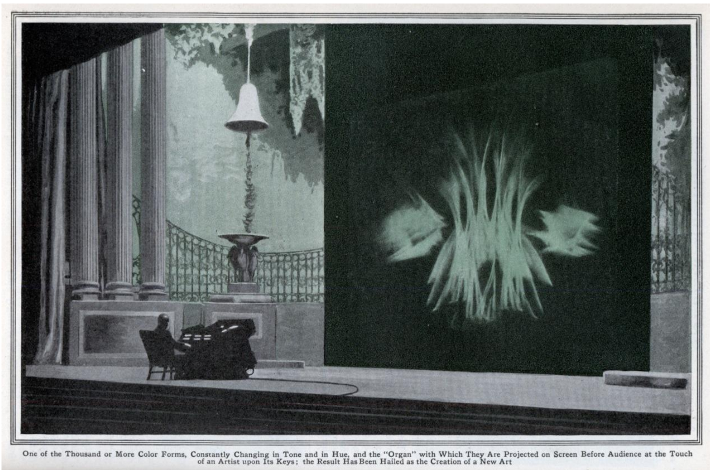



**Figure 3. Thomas Wilfred performing on a Clavilux (April 1924, image from (Bergstrom, 2011)).**

The first modern device that allowed live controlling the visual art from a musical input was Gordon Pask´s Musicolour system, whose first incarnation was created in 1953. The signal that controlled the visual performance was obtained by monophonic audio signal processing, where changes in amplitude (beats) were detected (Bergstrom, 2011).

During the 60´s and 70´s there was an explosion of activity in live visuals performance, which was very related to popular culture of the period. The great advancements in electronics and computer technology during those years led to the apparition of different devices, such as analogue video synthesizers, computer controlled laser shows, etc.

Although these devices supposed a revolution in the field, they were considered limited in terms of the possible outputs they could provide, i.e. in the case of laser shows, the output was limited by nature to a simple coloured line drawings. However, with the introduction of computer graphics in visual live performances, the number of possible outputs suddenly became enormous. The first computer system made with this intent was “Vampire”, created in 1970, at Bell labs by Max Matthews, as a successor of previously created music applications (“Groove”, “Music I”), since 1953.

At the present, a large number of such systems exist, the majority of them controlling visuals through data derived by performing beat and amplitude detection on the stereo mixdown of the music. Popular examples can be found on most personal computers, such as Nullsoft’s Winamp Advanced Visualization Studio and Apple iTunes Visualizer (Bergstrom, 2011).

A quite recent performance practice, in which music is frequently presented together with live visuals, is VJing (Video Jockey). VJing refers to the practice of playing and mixing (normally) pre-recorded videos, while adding effects to them in real-time. This practice is nowadays wining more and more adepts around the world, as more people is being attracted by the audio-visual art forms.

**Figure 4. Vj Vello Virkhaus accompanying the live group Red Hot Chili Peppers (Bergstrom, 2011).**

The previously mentioned inventions are only a few examples within a vast field in which many artists and scientists have experimented towards the interconnection of sound and vision. To learn more about the history of the field, read Bergstrom (2011) and Bain (2008).

## b) Relevant Concepts around Music Visualization

This section reviews some important concepts that often appear when talking about music visualization, and the present established practices around it.

### Synaesthesia

Although it was already described previously, a formal definition is offered here for synaesthesia:

> "*Synaesthesia (Greek, syn = together + aistesis = perception) is the involuntary physical experience of a cross-modal association. That is, the stimulation of one sensory modality reliably causes a perception in one or more different senses*" *(Cytowic, 1995)*.

Synaesthesia is a neurological condition. Subjects that have this condition, involuntarily experience certain sensory answer on more than one modality as a reaction to a particular stimulus. Synaesthesia comes in many different forms. The most common one is hearing induced vision (coloured hearing), where sounds trigger visual experiences. Other common cases include colour perception elicited by the reading of words or digits, hearing induced touch, vision induced smell and others. Normally, synesthetic mappings represent a one-way projection, which means that in a hearing induced vision case for example, not necessarily vision induced hearing is experienced. Another important characteristic about synaesthesia is that it is effortless, which in other words means that can not be consciously controlled any more than other more common perceptions (Ivry, 2000).

An important point is the idiosyncrasy of synaesthesia, which means that each synesthetic will have it´s own perception, and hence two persons with the same case of synaesthesia will not necessarily share the same synesthetic experiences. Unfortunately, this supposes a disadvantage from the point of view of music visualization, as the mapping between hearing and vision cannot be established as an objective "scientific universal rule" based on the experiences of synesthetic people.

### Audiovisual Composition and Multimodal Perception

An important concept, which could also be relevant to this project, is Audiovisual Composition:

> "*artistic form which takes as its starting point the cognitive actuality of multisensory audiovisual experience*" *(Grierson, 2005).*

Music and visuals get fuse to become a third art form, where one is not simply accompanying the other, but they are both experienced as an inseparable whole.

Arguments for the corroboration of this art-form are based on historical precedents in arts, where the premise of audiovisual composition has been of great importance, and research made in the areas of psychophysics and neuroscience, which supports the artistic suppositions of audiovisual compositions.

As precedents in art, we find Michel Chion´s widely established concept of synchresis:

> *"(...) the spontaneous and irresistible weld produced between a particular auditory phenomenon and visual phenomenon when they occur at the same time" (Chion et al., 1994).*

This theory, which defends that synchronized music and/or sound provides "added value", is part of the vital knowledge of sound for cinema.

Moreover, we also know from research that the human perceptual system is apt at detecting correlated stimuli across modalities, and fusing these into a single percept before their interpretation. Besides, it has been experimentally shown that there is a correlation between the amount of synchronisation and the perceived effectiveness of the combined audiovisual stimulus. Close correlation between visual and auditory musical events shapes a more effective experience in audiences.

These theories, and specially synaesthesia contributed to the argument that there is a strong indication that multimodal perceptions are not processed as separated streams of information, but are fused on the brain becoming into a single percept. On the contrary, a condition like synaesthesia would not be possible (Bergstrom, 2011).

**Visual Music**

A definition of visual music is provided as:

> *"Time-based visual imagery that establishes a temporal architecture in a way similar to absolute music. It is typically non-narrative and non-representational (although it need not be either). Visual music can be accompanied by sound but can also be silent"(Evans, 2005).*

This term refers to the use of musical structures in visual imagery, but also it is commonly used to denote the systems and devices that generate or transform sound and music into visual representations.

## c) Sound and Vision Connection

Through history, many theories have arisen aiming to interconnect sound and vision, focusing on different characteristics of both domains and potential similarities. Although some mappings seem to result intuitive to a vast number of people, such as pitch to spatial height connection, most of them are considered to be part of cultural knowledge, as pitch is represented with spatial height in traditional scores. This intuition in mappings is thus derived from acquired knowledge rather than from human innate perception. It is considered to be a lack of objectiveness in this kind of works, mapping visual and sound dimensions, as there is always implicit certain amount of the author´s personal interpretation. However, in this section some interesting theories and approaches are presented.

### Colour to pitches mapping

The beginning of synesthetic experience recreation in artistic works began with paintings in the 18th century, by the association between different colours and pitches in music. In 1704, Sir Isaac Newton came up with an original concept, throughout which he tried to create a one-to-one mapping of the seven notes on the piano to seven colours in the rainbow. Other artists and inventors, such as Castel, Bainbridge or Rimington, had different approaches on the appropriate sound to colour mapping. Figure 5 shows different approaches that have been used to map colour with notes (Bain, 2008).

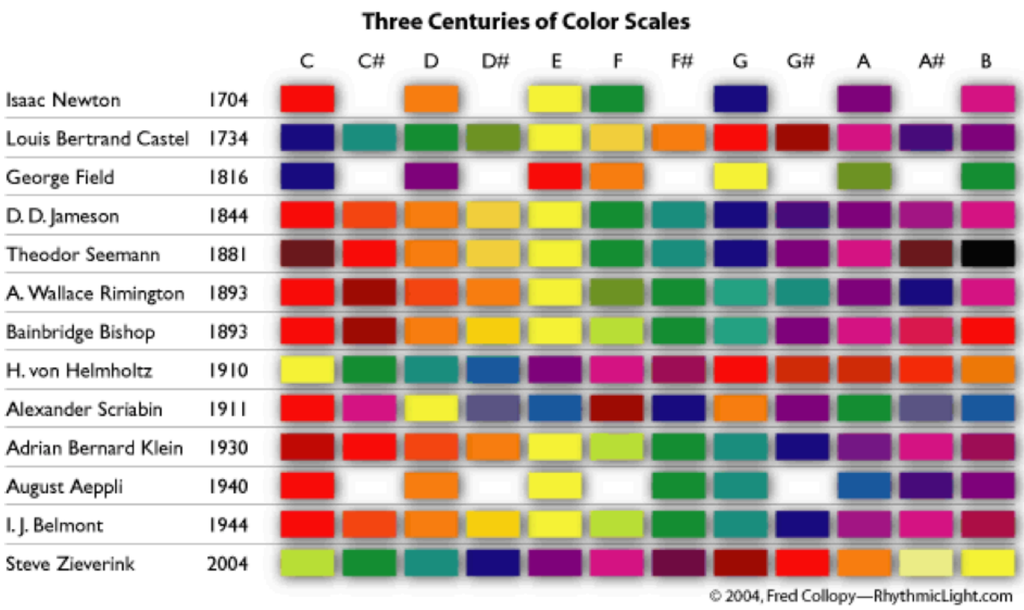



**Figure 5. Historical colour to notes mappings (Bain, 2008).**

### Visual Harmony: Differential Dynamics

The word harmony is nowadays used not only for defining the vertical dimension of music, but also the general sense of agreement and peace it evokes. Pythagoras defined this equilibrium mathematically, using whole number proportions that represented an ideal in music. Musical tunings based on integer ratios, denoted Just intonation, found their equivalents in visual arts, such as sculpture and architecture.

As it was shown before, many artists and scientists tried to find a connection of sound and vision through the properties of pitch, mapping it to spatial height or light and colors. However, although artists often speak of “color harmony”, attemps to directly map color to musical consonance and dissonance have largely failed. John Whitney defended that the mapping of music´s most basic parameters (pitch and loudness) failed to capture the expressive vision of great works of music, which, to him, depended more directly on multidimensional interplay of tension and resolution (Alves, 2005).

Whitney created a set of visualizations based on harmony creation not of color, space or musical intervals, but of motion. He discovered that a large number of elements in repetitive motions, with different integer ratios speeds, would demonstrate beautiful patterns at points corresponding to the same ratios that define musical consonances. He called this “differential dynamics”. Alves (2005) offers detailed explanations and representations of this concept.

## Timbre visualiation

Timbre characterisation is normally related to the spectrum of the sound, as the timbre depends on the energy distribution along the frequencies, i.e. the energy and location of the partials of a sound. Many studies in MIR, dealing with timbre classification, start with a representation of the power spectrum of a sound, using techniques such as Mel Frequency Cepstral Coefficients (MFCC).

Several visualization approaches can be found in state of the art related work. As an example, the River timbre visualization, which simulates a “river of sound” whose path, bounds, location and colour are controlled by several musical features (Siedenburg, 2009).

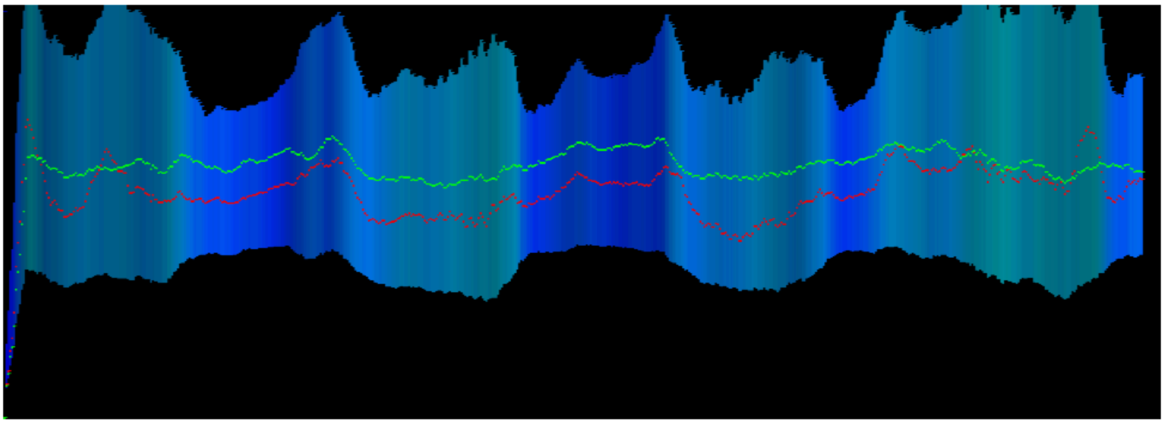


**Figure 6. "River" timbre visualization (Siedenburg, 2009).**

## d) Sound and Music Visualization

What humans perceive as sound is the propagation of an oscillation through a medium, normally the air. This oscillation consists on a change of pressure as a continuous process, time dependent. However, continuous-time signals cannot be processed by computer technology. For this reason, a process called sampling is required to convert the continuous signal into a discrete-time signal. Sampling a signal means to take a sample or the current amplitude value of that signal with a given periodicity (every fixed time interval). The values that discrete samples are going to take are defined by a process called quantization, which sets the current sampled amplitude value to the nearest available one. These available values depend on the word-length (in bits) used to describe a single value. Largest word-lengths lead to more precise values (Pramerdorfer, 2011).

Once sound signals are transformed to discrete-time signals, these can be processed by computers. The approach of extracting information from audio signals is called *content-based audio processing*, and it is one of the many disciplines within *Music Information Retrieval* (MIR) field.

Many features can be computed from audio signals. These features are normally classified by the music information levels, which were explained in the transcription section of this thesis (Figure 1). Another common way to classify these features is to divide them into time and frequency domain. Many temporal features can be directly computed from the time representation of the signal in frames, such as the mean, zero-crossing rate, loudness, RMS, etc. Likewise, many features can be extracted in the frequency domain, by first translating the time domain signal into it´s frequency representation, normally resulting on a spectrum or spectrogram. Some audio descriptors can be directly extracted from this representation, such as spectrum energy, energy per sub-bands, kurtosis, skewness, spectral centroid, etc.

The result of computing musical features from a sound can be used to control the visualizations, or in other words, the extracted information from the music descriptors is the data we communicate through the visualizations. At the present, there exist many software, tools and libraries for generating visuals from data. Some examples are described on the next sections.

## e) Visualization Approaches Classification: *Objective* & *Subjective*

When talking about music visualization, different approaches are comprised, which lead to different purposes. These may include, the simple representation of a waveform or a spectrum to visualize a signal into the time and frequency domain; the transcription of sound as accurate as possible, using scores or other notational system; and the artistic visualization of sound, which aim to create beautiful imagery to accompany music and create a sensory answer in the listener/viewer.

To distinguish between all these, I have created a classification to group the different visualization approaches by their intention when communicating information. I called these groups *objective* and *subjective* approaches. To judge whether a particular visualization is included in one group or the other, the intention of the data communication should be analysed: the accuracy and readability of the information.

I would like to remark that this classification has been created in order to simplify the explanation of the concepts and ideas I describe along this thesis, and does not pretend to become a "universal" classification rule, but a label to help naming things.

#### ***Objective* Music Visualization Approach**

The visualizations included in the *objective* approach group, aim towards the representation of sound and music information as accurate as possible. Within this group I include raw representations of waveforms and spectrums, and notational methods for the meticulous representation of musical features, such as traditional scores, MIDI piano-rolls, etc.

This kind of visualizations is intended to create knowledge, instead of an aesthetic experience for the user.

#### ***Subjective* Music Visualization Approach**

The visualizations included in the "subjective" approach consist of artistic/abstract imagery that accompanies music to create a stronger sensorial response in the listener/viewer. This means that the goal of this kind of systems is not intended to provide an accurate representation of musical information, as transcription/notational systems do, but to raise the artistic essence of music and the sensations it creates in the public by accompanying it with visual art.

This kind of visualizations is intended to transmit sensations to the user by the creation of an artistic aesthetic experience, instead of creating knowledge by reflecting accurate information of the music.

### f) Music Visualization Systems

As technology has progressed, so have the tools that permit the exploration of the relation between these art modalities. Moreover, people from many disparate fields face this challenge, from different perspectives and methods. This has lead to the development of new software tools to deal with audio and visuals, and the appearance of many systems that were created to visualize music or, beyond this, create audio-visual art expressions leading to a wider sensorial experience.

## Specialized software and tools

Nowadays there exist many software targeted towards the generation of graphics from data. Often, this has available functionalities to deal with audio too, which permits to work with different audio formats and perform signal processing techniques to obtain musical features from them. The visualizations can be controlled with the data derived from the musical features, and range from raw frame data that correspond to visualizing data as precisely as possible to the perceived music, (for example drawing the waveform of a frame), to *generative visualizations* that focus on producing beautiful images and impressive effects.

Here are a few examples of generative visualizations (created using Processing, a software oriented to the graphics generation). One of them is Audio-driven Landscape by Robert Hodgin, which creates a three dimensional landscape from the frequency distribution of the audio signal. This was obtained by smoothing frequency data and mapping time, frequency and intensity of the frequencies to the X, Z and Y-axis respectively, and the use of colour. The result is reflected on Figure 7.

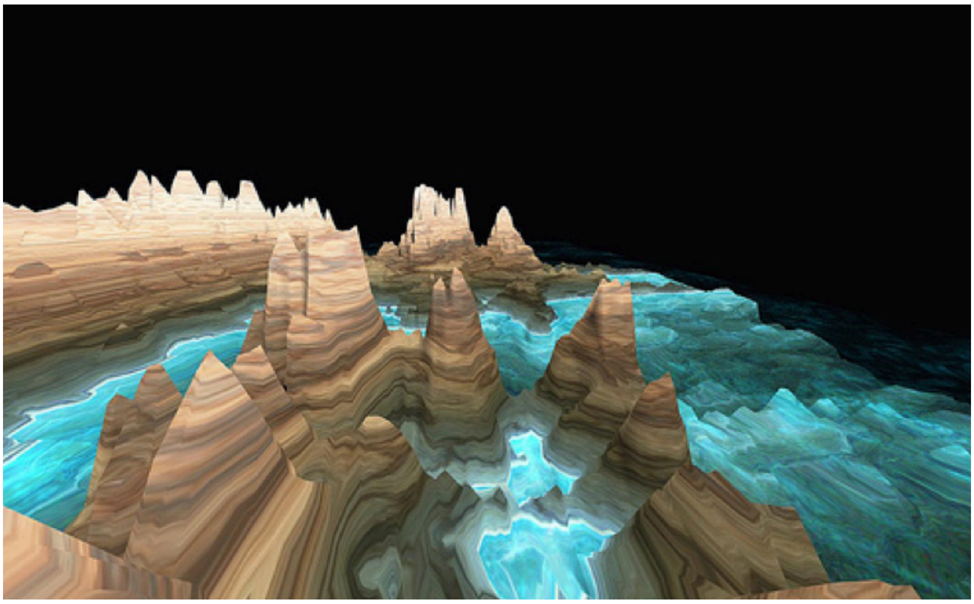
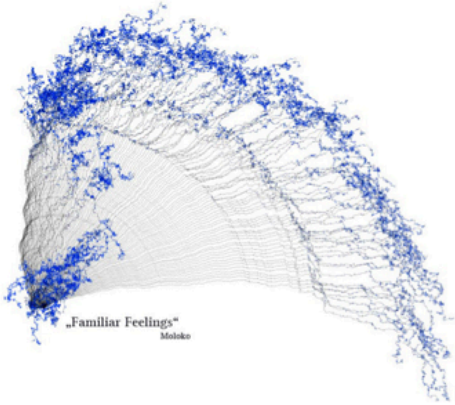


**Figure 7. Audio-driven Landscape by Robert Hodgin (Pramerdorfer, 2011).**

Other examples are Familiar Feelings by Moloco, which draws frequency bands independently, or Bubbles, which is based on a comparison between randomly generated values and RMS (Root Mean Square) values from the audio signal (Figure 8).



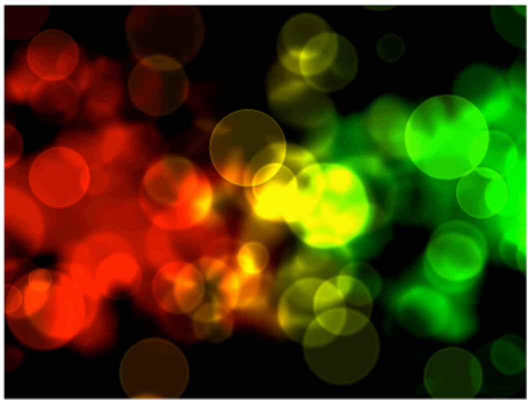



**Figure 8. Familiar Feelings by Moloko (left) and Bubbles by Pramerdorfer (rigth) (Pramerdorfer, 2011).**

### Soma

Soma is a system created in 2011 by Illias Bergstrom (Bergstrom, 2011) for the live performance of procedural visual music/audiovisual art. It was created with the purpose of improving the practice of this field´s art form and break through it´s main limitations, which included:

- Constrained mappings between visuals and music, which remain static along time and thus result limited in terms of complexity.

- Virtually no user interface existence, to control the performance of visual music/audiovisual art in real-time.

- The complexity of the process for preparing or improvising live procedural visual music/audiovisual performances.

- The limitation of collaborative performances in live visual music/audiovisual performance.

The author proposed a system that included solutions for the previously addressed problems. Figure 9 shows it´s main structure. The main functionalities were gathered within three software tools, named as the Trinity system: The Live Input Processor, responsible for gathering and processing audio and discreet control data from instruments, Mother, responsible for hosting visual synthesizers, and finally Mediator, which provides the functionality and user interface for controlling the mutable mapping.

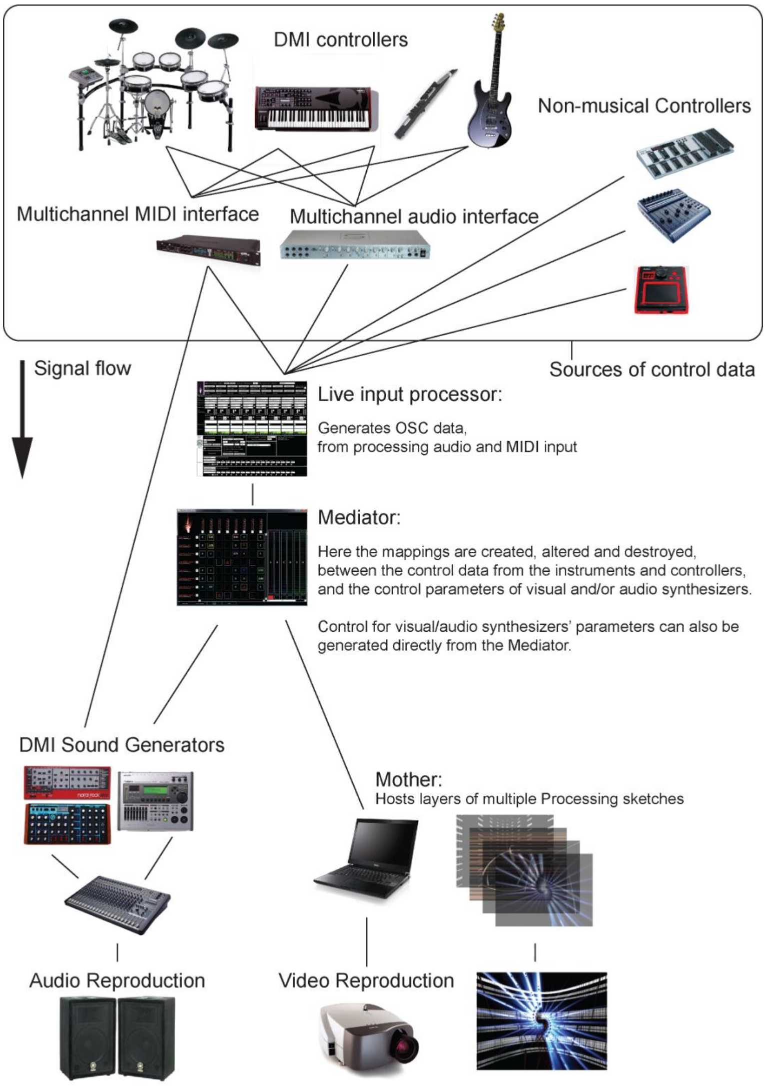



**Figure 9. Illustration of signal flow in Soma system (Bergstrom, 2011).**

With the develpment of the Trynity system, Soma covered the previously mentioned limitations, leading to the following contributions:

- Musical instruments are used as the primary source of control data for the performance. Musical gestures lead to much richer information than only beat events, amplitude and tempo from a stereo audio signal.

- Mutable Mapping: gradually creating, destroying and altering mappings between sound sources and visuals during the course of a performance.

- The user interfaces were provided in the Trinity system software.

- A reduction in the difficulty of preparing towards the performance of live procedural computer graphics is attempted, through readapting a programming language intended for artists.

From the previous contribution in conjunction, follows the inception of the art form of Soma. In Soma, correlated auditory, visual and proprioceptive stimulus is used to form a combined narrative. Soma builds on research findings that both performers and audiences are more engaged in a performance, when performers exhibit advanced motor knowledge, and when congruent percepts across modalities temporally coincide.

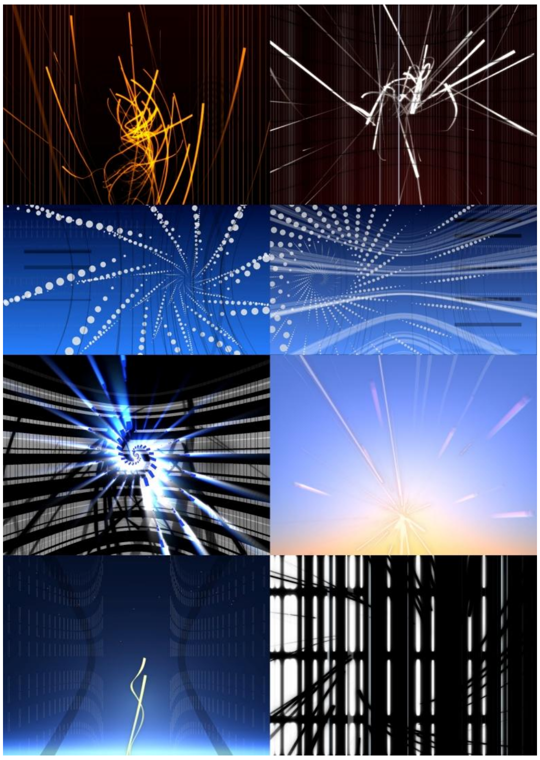


**Figure 10. Examples of visual outputs using the Trinity system (Bergstrom, 2011).**

## Magic Music Visuals

Magic is an application that allows to create dynamic visuals that evolve from audio inputs. It is though as a tool for VJing, music visualization, live video mixing, music video creation, etc. It allows to work with simultaneous audio and MIDI inputs, both pre-recorded tracks and live input signals ("Magic Music Visuals: VJ Software, Music Visualizer & Beyond," n.d.).

In this software, graphics are created with components called *modules*. These are connected together to form scenes. Visual modules can be linked to different audio and MIDI features. For an audio track, the overall amplitude of the signal, the amplitude per bands, pitch and brightness can be used. For a MIDI input, some features such as velocity, pitch bend or channel pressure can be used, among others. The graphic are produced using OpenGL which is a cross-language, multi-platform application programming interface (API) for rendering 2D and 3D vector graphics. The software also provides the functionality of creating new modules to developers.

The next figure shows an example of the graphical user interface of the software. The boxes are the modules that have different functionalities and are interconnected to create scenes.

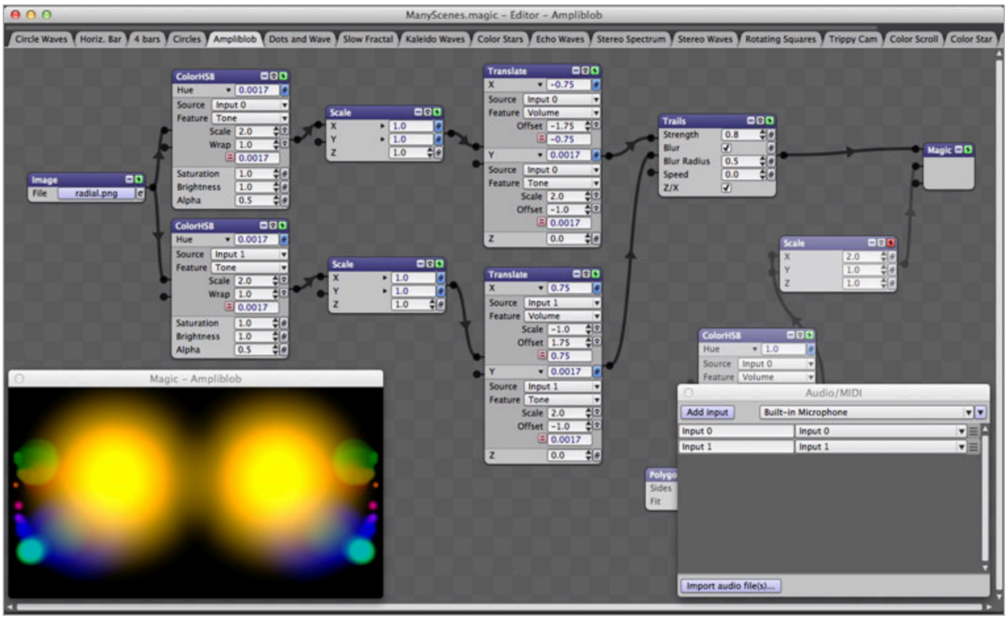



**Figure 11. Magic Music Visuals graphical user interface.**

## Guitar Visualization Systems

The previously presented systems (Soma and Magic) are examples of systems that permit the visualization of music, aiming to create beautiful imagery as an aesthetic experience for the viewer (*subjective* visualization approach).

Also, this kind of systems are sometimes more generic as they are designed for controlling the visuals from a group of DMI (Digital Musical Instruments) controllers, or directly from a stereo audio signal representing the mix of all the instruments that compose the piece.

In contrast, the systems described in this section have a slightly different approach, in which the aesthetic is important when creating graphics, but the aim is to accurately represent musical information so that the piece can be interpreted. In other words, these systems could be understood as performance instructors, rather than visual art generators. Moreover, these systems are focused on guitar, which leads to a more detailed characterisation of it´s particular sound, in contrast to the extraction of musical information from the whole stereo mix of a piece, composed of several instruments. Thus, following the classification criteria exposed before, these systems could be located within the *objective* visualization approach.

One of the possible reasons that contributed to the popularity of these systems was the release of Guitar Hero ("Guitar Hero Official Site," 2005), a videogame that appeared in 2005, which aimed to recreate the experience of playing guitar and make it available to everyone as a game. It consisted of a DMI guitar-shaped controller through which music was "interpreted" by the player, who was guided by the instructions that appeared on the screen. These instructions consisted of the notes of a particular song, presented over time. So, with the original song´s backing track sounding, the aim of the player was to press buttons on the guitar controller in time with musical notes that scroll on the game screen.

Another game called Rocksmith ("Rocksmith," 2011) was released in 2011. This game followed the main idea of music performance instruction (as Guitar Hero), but with an essential difference: a real guitar was used instead of a DMI controller. The idea behind the game was to be able to use any electric guitar, so it was approached as a method of learning guitar playing. The game offered a set of songs, for each of which a performance instruction was presented based on the notes that had to be played along time. Then, some feedback about the performance quality was given to the user.

There are other systems that follow the same approach of guitar music visualization as notation, to give the user the necessary instructions to reinterpret a particular piece. Some examples of this are GuitarBots and Yousician ("Yousician," 2010). These systems provide an easier way of learning to play guitar by helping the user with visual instructions about what to play.

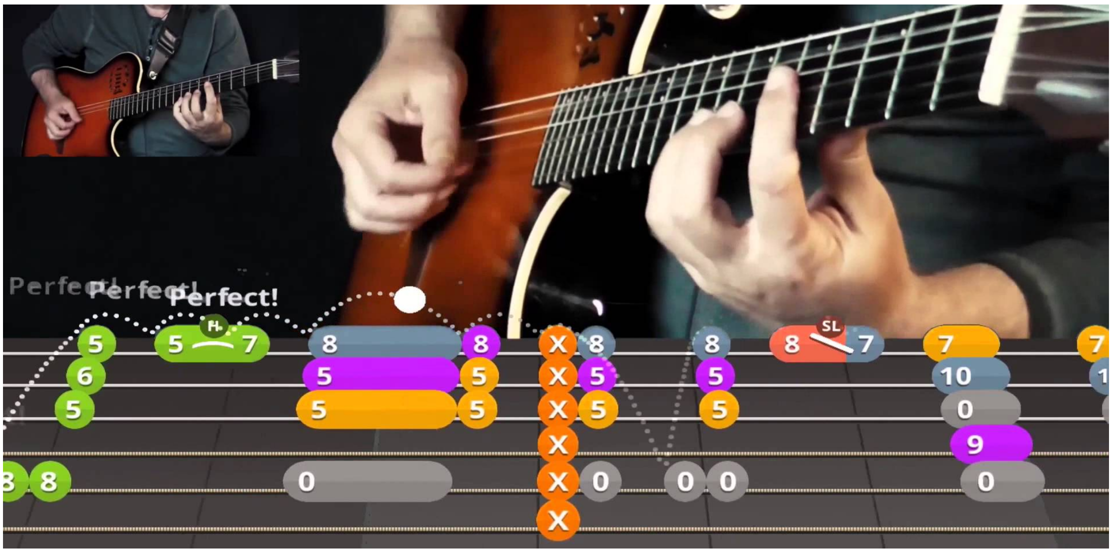



**Figure 12. Yousician guitar playing.**

## 2.3 A Music Visualization System for Hexaphonic Guitar

The idea Klapuri (2004) stated of music transcription as any symbolic notation transmitting sufficient information to interpret the piece supposes a key point for this project. It will act as a merging point between music and visuals, where the transcribed music will be represented using different visualization approaches.

Traditional scores are maybe the best way to transmit music information because it´s effectiveness and, of course, because it is probably the method we are most familiarized with. However, the notation/visualization methods proposed here are pretended to offer alternative perspectives for guitar playing, offering accurate musical information about the guitar performance, but presenting it in alternative visual ways.

This research project does not pretend to give an absolute answer to the problem of matching music and visuals and to go beyond the barrier of subjectivity and "arbitrarism" in the mappings between visuals and music, but to study how could alternative visualizations better fit the users' perception.

The goal of the system proposed in this project is to analyse how different visualizations, based on both the *objective* and *subjective* visualization approaches, could result into a richer experience of music, in the sense of adding visual stimuli to accompany the music and representing the musical information that best characterises guitar performances.

The distinction between *objective* and *subjective* visualization approaches would lead to different applications of such a system. For example, developing an *objective* visualization in which information about music structure, such as harmony, is collected, could suppose a tool for learning guitar playing for the musician. On the other hand, the *subjective* visualization approach in which an aesthetic experience is created for the public could be used as a tool for live shows, in which music is generating visual effects that evolve with it.

The system we propose is thus focused on conventional guitar, and deals with audio signals from which musical features are extracted. Monophonic transcription techniques are used, hence to be feasible the hexaphonic guitar construction is needed. Furthermore, we took into an account some of the limitations addressed by Bergstrom (2011) in the Soma system. Our system is focused on a particular instrument leading to more detailed control of the musical features in comparison to dealing with a stereo mix signal. Also, we focus on giving the user manageability and flexibility to configure the mappings between sound and vision through a GUI (Graphical User Interface), working in real-time during the guitar performance.

# 3 METHODOLOGY

The development of this project has been divided into two phases, based on an iterative process of design, development, evaluation and feedback. These two phases used different approaches and resources, and hence produced two different prototypes of the system.

In this section, the steps followed towards the implementation of the project are described, by firstly analysing the tools and resources that characterises each of the prototypes, and then explaining the way in which these are used and mixed together.

## 3.1 Design: Tools & Resources

A general overview of the tools and materials employed for the development of the prototypes is offered in the next sections.

### a) Design of the First Prototype

The First Prototype achieves hexaphony using a classical guitar. The sound of each string of the guitar is separated and sent to the computer through an independent channel of an audio interface. Then, the audio signals are processed using Essentia library, through which musical features are extracted to characterise the sound. This information is then sent to Processing, graphic programming environment, to create the visualizations.

**Essentia**

Essentia (Bogdanov et al., 2013) is an open-source C++ library for audio analysis and audio-based music information retrieval. The design of this library is focused on the robustness of the music descriptors it offers, as well as on the optimization in terms of the computational cost of the algorithms. Essentia offers a great collection of algorithms, which compute a variety of low-level, mid-level and high-level descriptors useful in the MIR field (Figure 2). Moreover, it also provides additional tools for working with audio input/output and processing, and gathering data statistics.

This library is cross-platform and it is also wrapped in Python to facilitate the usage to a wider number of users, who may be familiar with matlab/pythton enviroments. Essentia suposes a powerful tools that collects many state-of-the-art techniques for the extraction of music descriptors and optimized for fast computations on great collections.

### Processing

As mentioned before, Processing is a programming language and development environment based on Java, as well as an online community. It has promoted software literacy within the visual arts and visual literacy within technology since the early 2000´s. Processing supposes a professional development tool nowadays, in a free open-source cross- platform software. There are many contributors who share code and build libraries, tools and modes to extend the possibilities of this software. At the present more than a hundred libraries exist to facilitate computer vision, data visualization, music composition, networking, 3D file exporting and programming electronics. For further information about this software see ("Processing," 2001).

### Hardware: transduction based on piezoelectric sensors

Transduction is the process for which a type of energy is transformed into another type. In this case, the process refers to converting the vibration of a string of the guitar (a mechanical force) into an electric signal, to be transmitted to the audio input device of the computer. For achieving this, piezoelectric sensors are used.

These sensors work with piezoelectricity, principle which states that electric charge is accumulated in certain solid materials in response to applied mechanical stress. Thereby, one of these sensors is able to transform the mechanical force that it experiments by the vibration of the string into an electric current representing this vibration, i.e. these sensors act like small microphones capturing the sound produced by each of the strings.

## b) Design of the Second Prototype

The Second Prototype characterises for using an electric guitar. In this case, a divided pickup is used to achieve hexaphony, which captures the sound of each string independently. Aditionally, a special circuit is needed to adapt the output of the divided pickup to the inputs of an audio interface, used to send the signals into the computer to be processed. In this approach, the musical feature extraction task is performed using Pure Data programming language. Also, the visualization generation is performed within this environment, by using Pure Data´s GEM graphics environment.

### Pure Data

Pure Data (also known as Pd), is an open source visual programming language, developed during the 90s by Miller Puckette. It allows the user to create software by manipulating program elements graphically rather than textually specifying lines of code. Pd is a so-called data flow programming language, where software called patches are developed graphically. Algorithmic functions are represented by objects, placed on a screen called canvas. Objects are connected together with cords, and data flows from one object to another through this cords. Each object performs a specific task, from very low-level mathematic operations to complex audio or video functions. Pd is a major branch of the family of patcher programming languages known as Max (Max/FTS, ISPW Max, Max/MSP, jMax, DesireData, etc.), originally developed by Miller Puckette at IRCAM (Puredata.info, 2016).

**GEM library**

GEM is the Graphics Environment for Multimedia. Written by Mark Danks, it was created to generate real-time computer graphics, especially for audio-visual compositions. GEM is a collection of externals which allow the user to create OpenGL graphics within Pd (Puredata.info, 2016).

**Hardware: transduction based on divided pickup**

As stated before, this prototype uses a conventional electric guitar. To transform it into hexaphonic, the Roland GK-3 divided pickup is used to separate the sound of each string. Furthermore, the output of this pickup (13 pin DIN cable) is adapted via a special circuit to the audio interface input (6 different input channels, Jack connectors). The construction of this circuit, the Breakout Box for the Roland GK system, is detailed in ("Unfretted - Fretless Guitar Resource.," n.d.).

## c) Advantages and Disadvantages of the Prototypes

Comparing the two prototypes, we identified some pros and cons each of them offers. From the hardware point of view, the first prototype results very affordable to implement, as the materials needed for it´s construction, i.e. piezoelectric sensors, are cheap and easy to find. However, although the circuit is simple and easy to build, the performance results poorer than the second prototype´s. A particular sensor captures the sound of it´s string, but also the sound of the neighbour strings. These are present in the signal at a lower intensity, but a threshold should be established to limit the dynamic range of the captured sound.

The second prototype's hardware performance results more robust, as the Roland GK-3 pickup separates better the sound of each string. However, the construction of the circuit to adapt the pickup output to the audio interface input is more complex. Also, the pickup and the materials for building the circuit suppose a bigger expense.

From the software implementation point of view, the second prototype performs better as both the music feature extraction and graphic generation is done within the same program (Pure Data), which results into a decrease of latency time.

## 3.2 Development

This section describes the development of the project in terms of hardware and software implementation. It details how the previously described tools are used and mixed together to build the system.

### a) Development of the First Prototype

**Hardware**

As explained in the previous section, some hardware is needed to transform a traditional guitar into a hexaphonic one. It consists on six piezoelectric sensors, six 1⁄4" TS jack connectors and twelve wires to interconnect them. With this material a circuit was build to capture the signal from each string independently and send it through a cable to the computer ́s audio input interface.

The construction scheme is reflected in Figure 13. Each piezoelectric sensor is welded with a jack connector as shown in the scheme. The tip of the jack (the shortest part in the end) is connected to the white inner circle of the piezoelectric, and transmits the signal. The sleeve of the jack is thus connected to the golden surface, being the ground of the circuit.

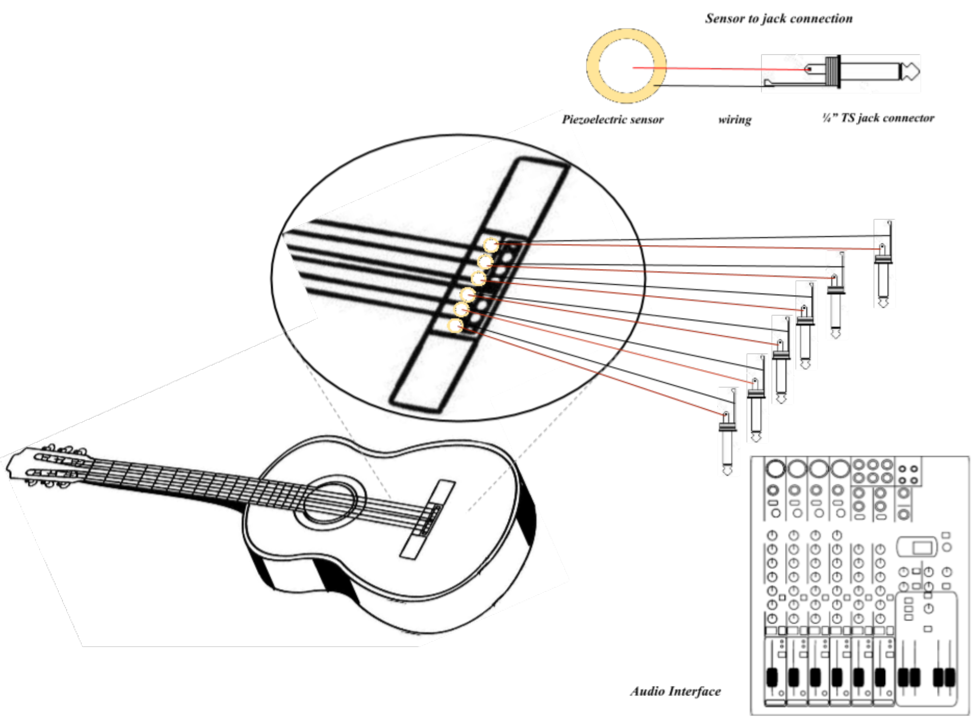



**Figure 13. Hexaphonic guitar construction scheme (First Prototype).**

Afterwards, each sensor is cut and placed between the string and the wood of the bridge of the guitar, place where we found the signal is better captured. Finally, the output jack connectors are plugged into different channels of the audio interface, so that the signals could be independently processed.

**Computing**

The software implementation was done on different steps and using several tools. Figure 14 shows an overall view of the process followed to compute the guitar sound visualization. This process is performed in real-time, which means that the visualization is produced while the user is playing guitar.

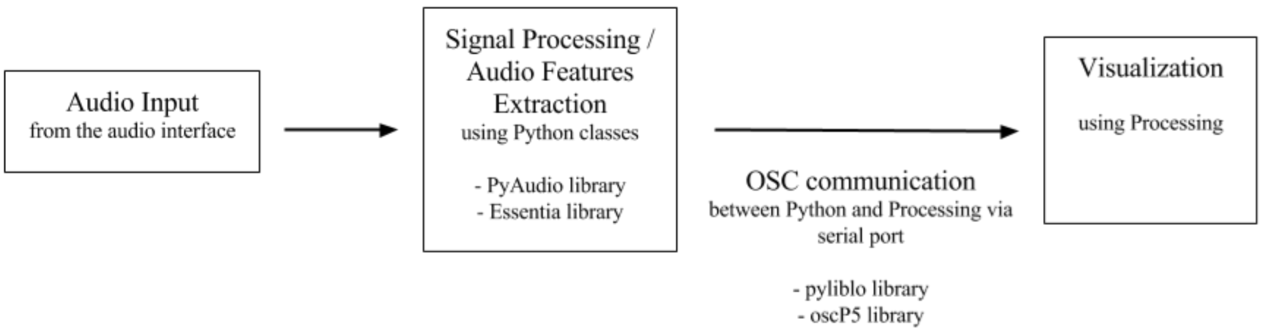



**Figure 14. Software implementation steps (First Prototype).**

**Feature Extraction**

At this step, the audio signals from the different channels of the audio interface are processed. To perform this task, a group of Python classes were created to manage the input audio buffers, perform the music features extraction with Essentia and send this information to a serial port via OSC communication. Simultaneously, this serial port is read from Processing to obtain the data to build the visualizations.

To deal with the audio, a library called PyAudio was used with Python programming language. This library provides Python bindings for PortAudio, which is a free, cross-platform, open-source audio I/O library. Using this library, an input buffer for audio management was created.

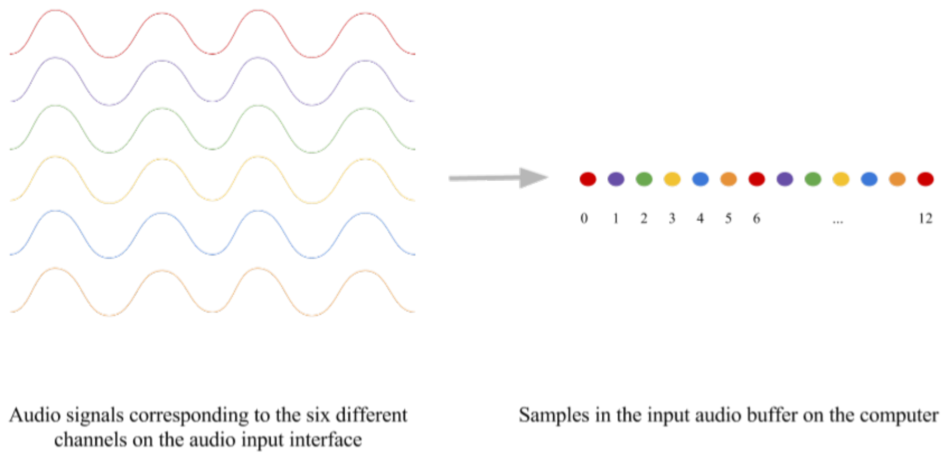



**Figure 15. Audio signals sampling process.**

As we can see in Figure 15, audio samples corresponding to the different signals on the six channels appear interlaced. Our Python classes separate the samples corresponding to the different channels, creating six different audio frames, which are then computed with algorithms from the Essentia library to obtain the desired descriptors. At the current implementation, the used algorithms are PitchYinFFT and Loudness. Past implementations included RMS and HPCP, but these were discarded to avoid computational extra-time.

### OSC communication

The extracted features, result of the Essentia algorithms computation, are then sent to a serial port using PyLiblo library, which is a wrapper for the liblo OSC library. This LibLO library is an implementation of the Open Sound Control (OSC) protocol.

In Figure 16 we can see an example of the data transmitted by OSC to Processing. For each channel (*C1, C2, etc.*), fundamental frequency, chroma note and RMS value is transmitted here. *This image corresponds to an old version in which HPCP was included and RMS was used instead of Loudness.*

```
C1 .: 98 , 5 , 1.42944896221   /\   C2 .: 99 , 10 , 0.759450376034
C1 .: 98 , 10 , 0.464081317186   /\   C2 .: 99 , 10 , 0.7594503760
C1 .: 98 , 10 , 0.464081317186   /\   C2 .: 97 , 5 , 0.47081434726
C1 .: 98 , 10 , 0.46189007163   /\   C2 .: 97 , 10 , 0.45322278142
C1 .: 96 , 10 , 0.382231861353   /\   C2 .: 97 , 10 , 0.4532227814
C1 .: 96 , 10 , 0.382231861353   /\   C2 .: 99 , 10 , 0.4290815293
C1 .: 102 , 11 , 0.444133549929   /\   C2 .: 100 , 10 , 0.58688366
```

**Figure 16. OSC data transmisión example.**

The class programmed in Processing uses another library called OscP5, which allows to continuously read a serial port and process incoming OSC messages. These contain the audio features data that is used to control the created visualizations.

## b) First Prototype Visualizations

For the first prototype, we have used a 2D plane for representing the notes played by the guitar. First, the strings have been distributed along the X-axis of the plane, which means that the notes played in a particular string will always be represented in the same vertical line.

From the fundamental frequency of the notes we obtain pitch. The pitch is mapped to the spatial height where notes appear, corresponding to the Y-axis. For example, if we play a D note followed by a E note (in the same octave), the second one will appear above the first one.

Notes are represented with circles that appear on a particular point of the two-dimensional space depending on the string that is played and the pitch height. The size of a circle varies according to the loudness of the note, i.e. higher amplitude of the signal would lead to a bigger circle.

Moreover, the name of the note (C,D,E,F,G,A,B and sharps), appear inside the circle. In the beginning this was done with the result of the HPCP algorithm, but later I realized that with the pitch information was enough, by creating a map of the fretboard fundamental frequencies (in standard tuning).

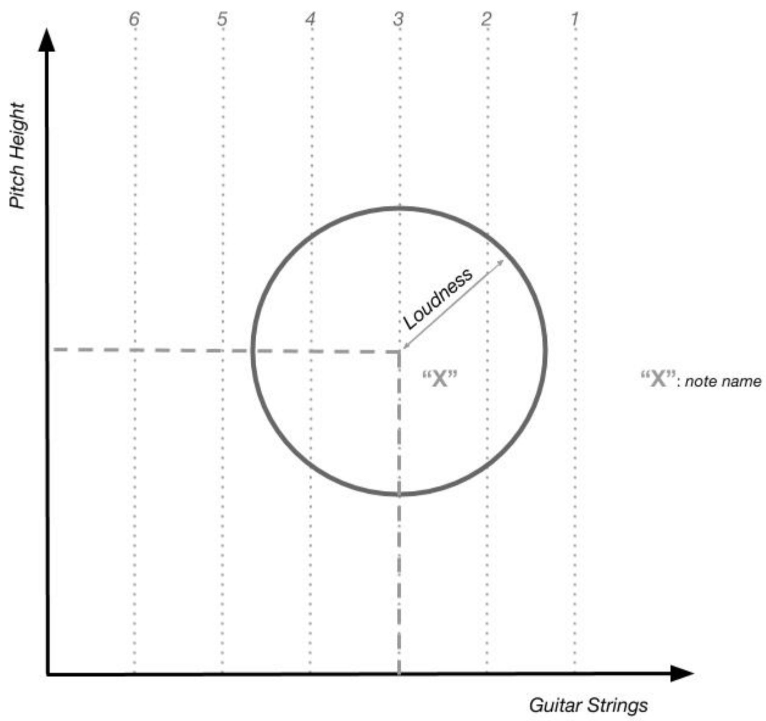



**Figure 17. Mapping scheme in visualizations (First Prototype).**

Lastly, each note is mapped to a particular colour. Although there is no clear scientific evidence on the accuracy of this mapping, I wanted to try different visualizations using colours as many artists and inventors have done along history. Besides, adding colours leads to more impressive and attractive visualizations, important factor when the aim of the visualizations include creating an aesthetic experience for the user.

Colour mapping was based on the range of visible frequencies spectrum. We selected the lower frequency colour (red) and mapped it to the lower frequency a guitar can produce (in standard tunning: E note). The other notes are sorted along the colour range, thus fitting all the visible frequency range to an octave.

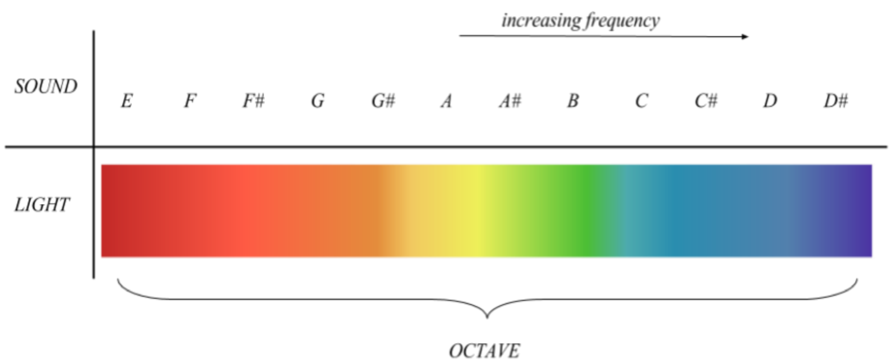



**Figure 18. Note to colour mapping.**

In the following figure, an example of the visualization interface is shown (Angulo, Giraldo, & Ramirez, 2016).

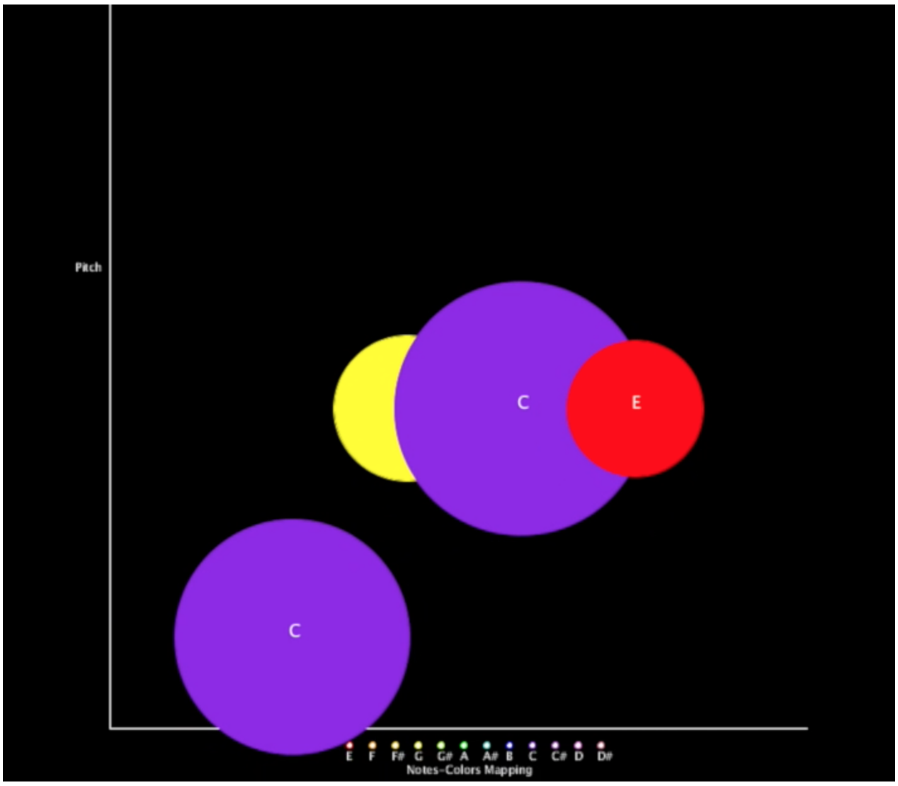



**Figure 19. Visualization GUI example (First Prototype).**

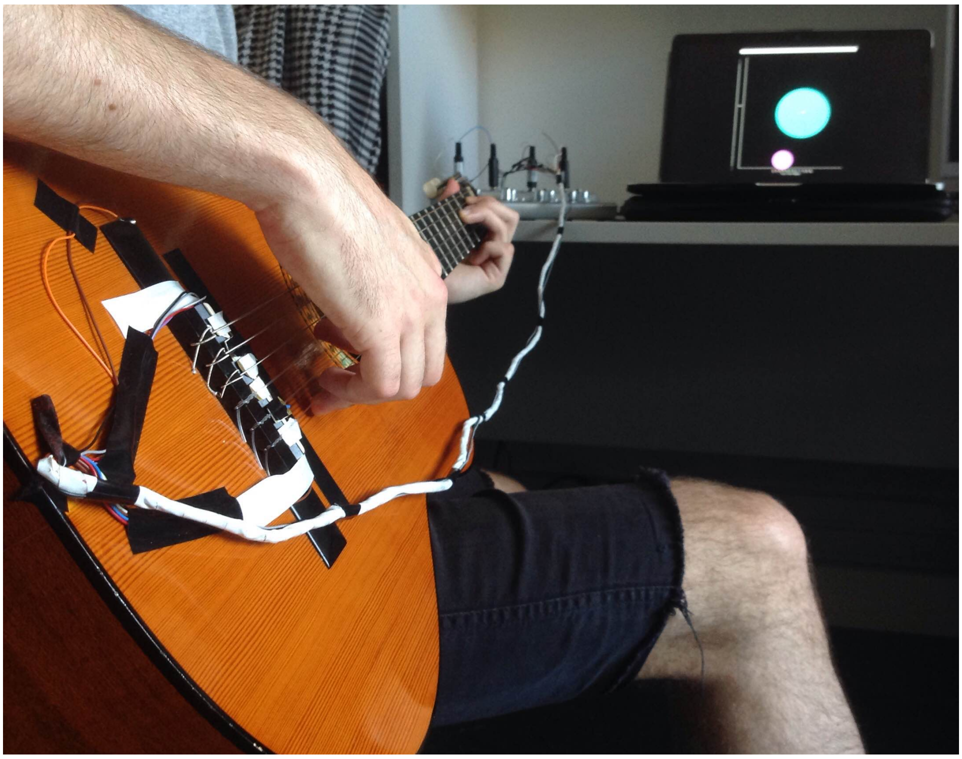


**Figure 20. First prototype setting.**

## c) Development of the Second Prototype

### Hardware

In the second prototype, sound transduction is done by the Roland GK-3 divided pickup, which is able to separate sound from each string. However, the output of this pickup consists of a 13 pin DIN cable. To introduce the sound signals into the computer, an audio interface is needed. So, it is necessary to adapt the pickup output so that the sound of each string can be inputted to the computer through an independent input channel of the audio interface. To achieve this, a Breakout Box circuit was built, following the methodology presented in (“Unfretted - Fretless Guitar Resource.,” n.d.). Basically, it consists of welding the pin (DIN cable) corresponding to particular string to a jack connector. Also, two batteries are needed so that the pickup can be fed from the circuit. The final circuit results in a box in which the 13 pin DIN cable is inputted, and 6 separate Jack connector cables are outputted. Figure 21 shows how this is settled. All the materials and tools needed for it´s construction are detailed in the referenced documentation.

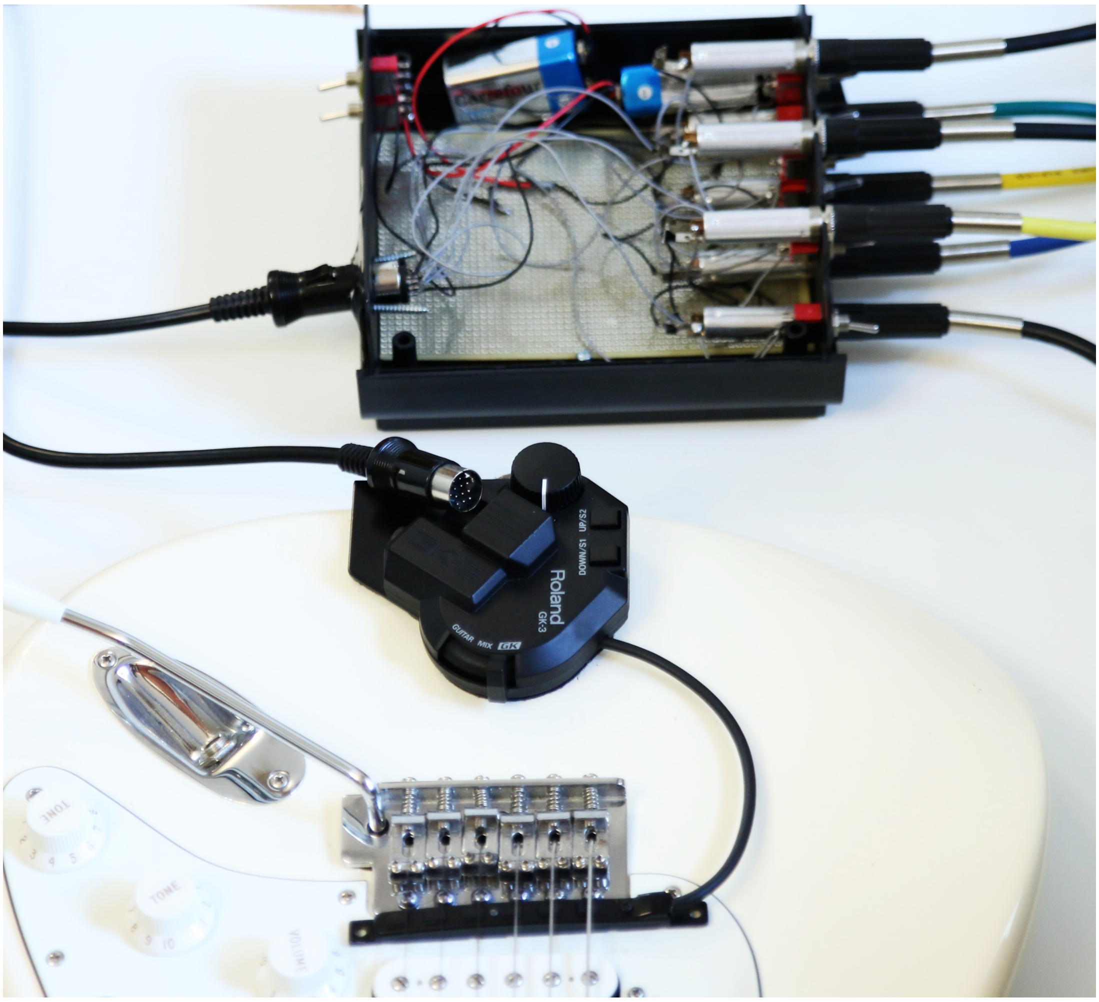



**Figure 21. Roland GK-3 and Breakout Box setting.**

**Computing**

Once we have the audio input signals, all the processing towards the visualization generation is done using Pd. A Pd patch has been created in which all the controls of the system are included, giving the user the option to configure several parameters affecting both the signal processing and visualization generation.

**Feature Extraction**

Each signal is independently processed, following the hexaphonic concept in which each string is considered as a monophonic sound source. For each string, the processing carried out is summarized into the next steps:

- Onset detection: the algorithm is by default set to be "inactive". It only runs when a note onset is detected, using Pd´s fiddle object. This is a way of saving computational resources and keeping it simple.
- Envelope detection: once an onset is detected, the envelope detector starts to run for tracking the energy of the signal. This is used to set the note offset by the signal´s energy level. A threshold parameter is offered on the GUI to the user, so it can be adjusted to respond to different levels. Also, an input gain parameter can control the signal energy level. Once the signal has gone under the threshold, the note offset is considered and the algorithm returns to it´s "inactive" state.
- Pitch detection: this is done by getting a small frame of the signal once the onset is detected, and some milliseconds after it, to avoid the initial fluctuation (noise) of pitch during the note´s attack. These time intervals, both the initial delay time and the amount of time the signal is listened by the algorithm to estimate the fundamental frequency can be adjusted by the user too (although there is a configuration settled by default in which the algorithm works good). For pitch detection SNAC (Specially Normalized AutoCorrelation) has been used, through Pd´s helmholtz object (Katja Vetter, 2012).
- Note classification: after the pitch detector has estimated the fundamental frequency of the note, some information is extracted from it. First of all, as the sound from each string comes from a different input channel, it is easy to know which string has produced a given note. A frequency map has been elaborated for the standard tuning of a conventional electric guitar, so once the fundamental frequency is obtained, it is easy to deduct the fret that has been pressed on it. Moreover, a numeric identifier has been created for each note, i.e. the semitones (C, C#, D, D#, E … B) correspond to (1, 2, 3, 4, 5 … 12). This will be useful for working with harmonic information of the music. Also, from the frequency map the octave of a given note is extracted, as electric guitar (in standard tuning) typically contains from E2 to C6. When a note is played, the octave number is

identified from it´s position on the fretboard. So summarizing, when a note is played, after the pitch estimation, the fret number, note's numeric identifier and octave number are also given.

The previous extracted features, i.e. pitch, energy, onset, are basic for the characterization of guitar sounds from a transcription perspective. These features are computed for each of the strings and permit the accurate identification of the instantaneous played notes at a given moment.

## d) Second Prototype Visualizations

In this case, different kinds of visualizations have been designed, for different purposes.

***Objective* visualization approach: Guitar Fretboard**

This visualization aims to accurately represent the guitar performance, and reflect useful information for the musician. It consists of a fretboard that mimics what is actually happening on the real guitar. When the algorithm is "inactive", the strings in the visualizations are drawn grey and static, as no note is being played (Figure 22). When an onset is detected, the string turns white, and the line reflects the waveform of the signal for that particular string. The string vibrates from the fret that is being pressed (the one that corresponds to the generation of that note) to the bridge of the guitar.

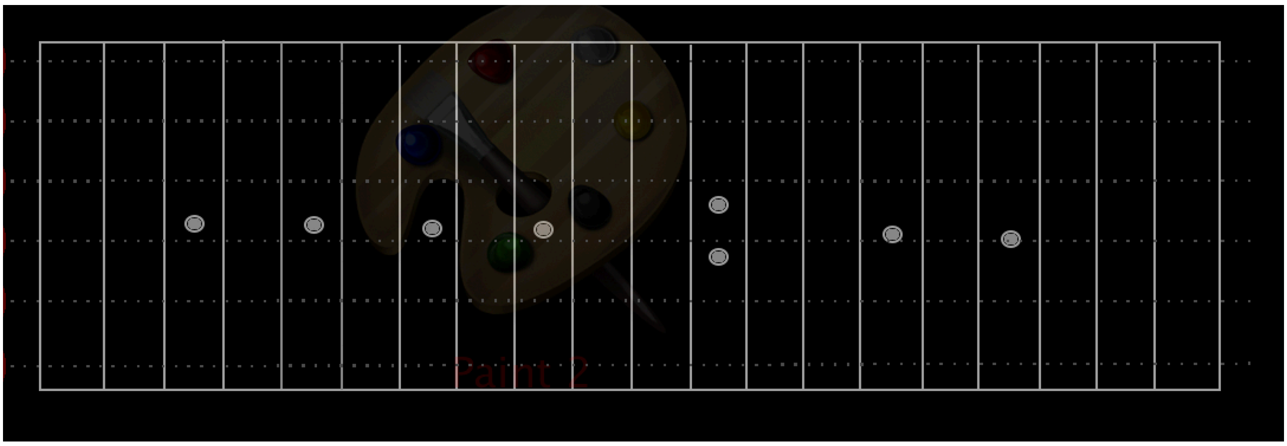


**Figure 22. Fretboard inactive.**

When a note is played, it´s name and octave are presented within the pressed fret. Also, a red circle appears to remark the position of the note on the fretboard, and it´s size is mapped to the note's energy (loudness).

Additionally, a chord dictionary has been created to indicate the chord that the played notes in a given moment form. This is done using the notes numeric identifier, to create an array of the played notes in a given moment. Then, the contour of this array is compared to chord templates (Figure 23). When the array fits into a template, the chord is identified, as well as the root note of the chord. In Figure 24, an example is shown of a F major chord being played.

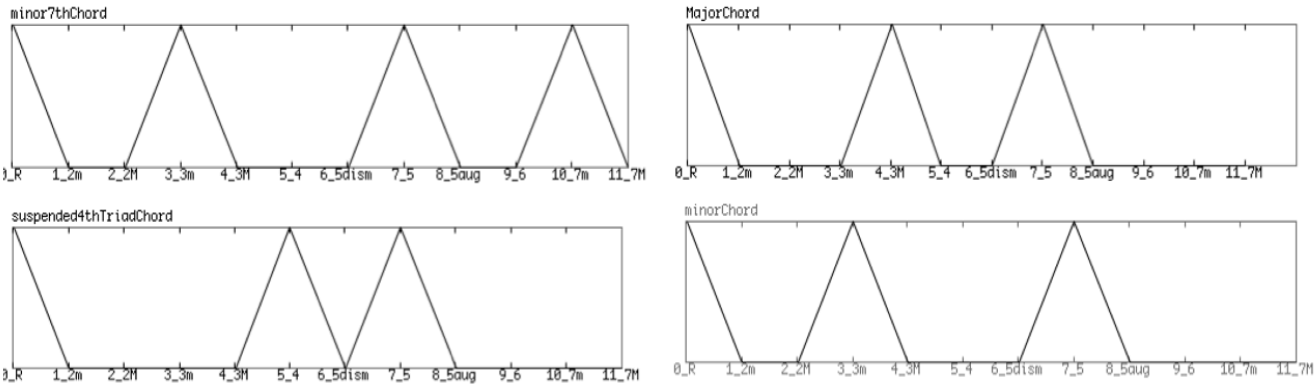



**Figure 23. Chord dictionary template examples.**

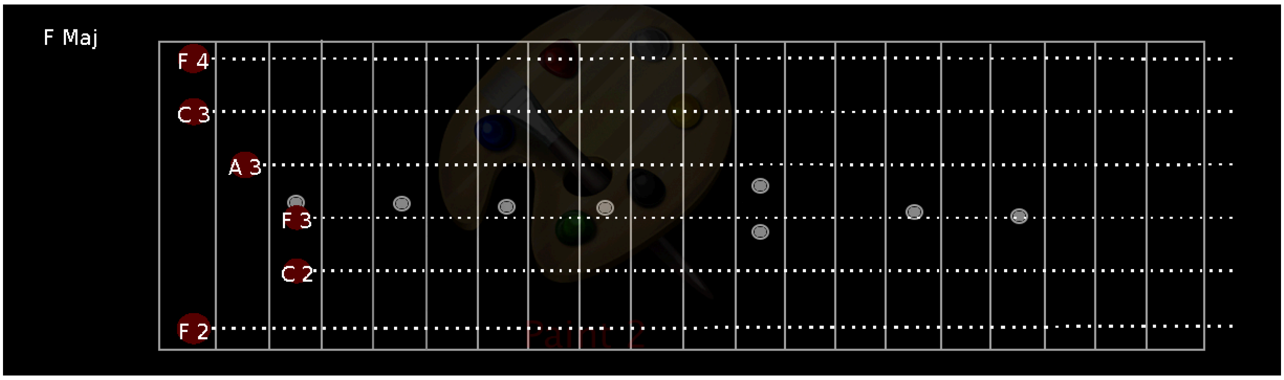



**Figure 24. Fretboard visualization.**

I consider this visualization to be included in the *objective* visualization approach, as it aims to accurately represent musical information so that it can be easily interpreted by the user. The aim of this approach is not intended to create beautiful imagery to transmit an aesthetic experience to the user, but to give useful information and feedback to her/him.

This works in real-time, which means that the visualizations are created as the musician is playing. From this point of view, this visualization approach could suppose a tool for the musician, for people who want to learn guitar playing and harmony. Apart from the chord dictionary, further information could be added, such a key detector, a visual guide for scales to show the user which notes "can" be played in which contexts, etc.

### *Subjective* visualization approach: Mapping Configurator

For the subjective visualization approach, we have developed a Graphical User Interface to bring the user the opportunity of configuring the mappings of the visualizations. Starting from the visualizations created for the first prototype, more elements and parameters are added to widen the possibilities of the visuals generation.

The GUI was designed so that the user could configure the mappings in real-time during the guitar performance. An audio looper was developed so that the guitar music could be pre-recorded and endlessly played, and hence offer the user the capability to create the visualization for a particular piece, configure it and then play music with her/his own parameter configuration.

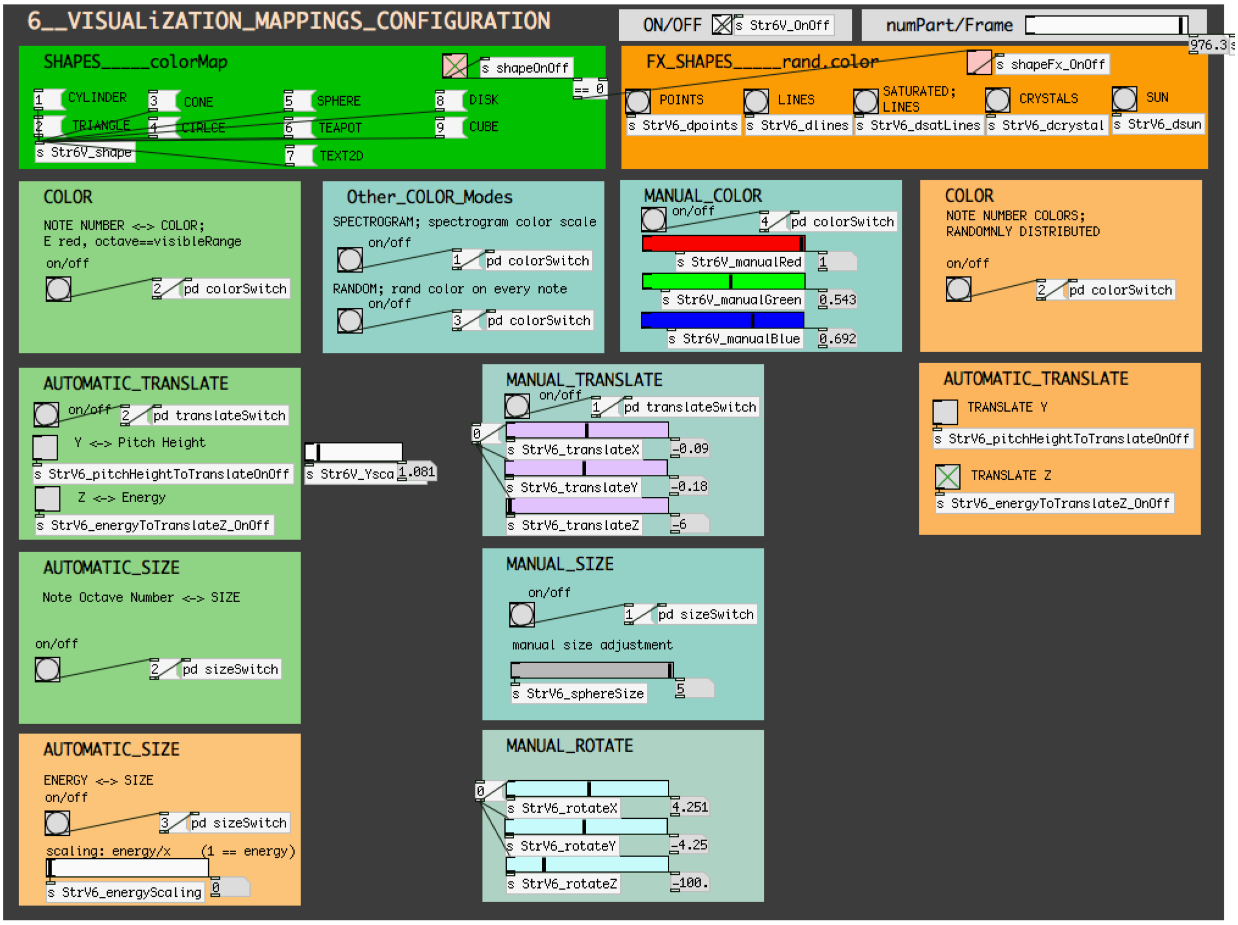



**Figure 25. Mapping Configurator Graphical User Interface.**

Through the Mapping Configurator (Figure 25), the user can adjust each string's visualization parameters. First of all, the object for representing notes, which in the first prototype was a circle, could be selected from a range of 2-Dimensional (triangle, square, circle, etc.) and 3-Dimensional objects (cone, cube, sphere, cylinder, and further effects as shown in Figure 26).

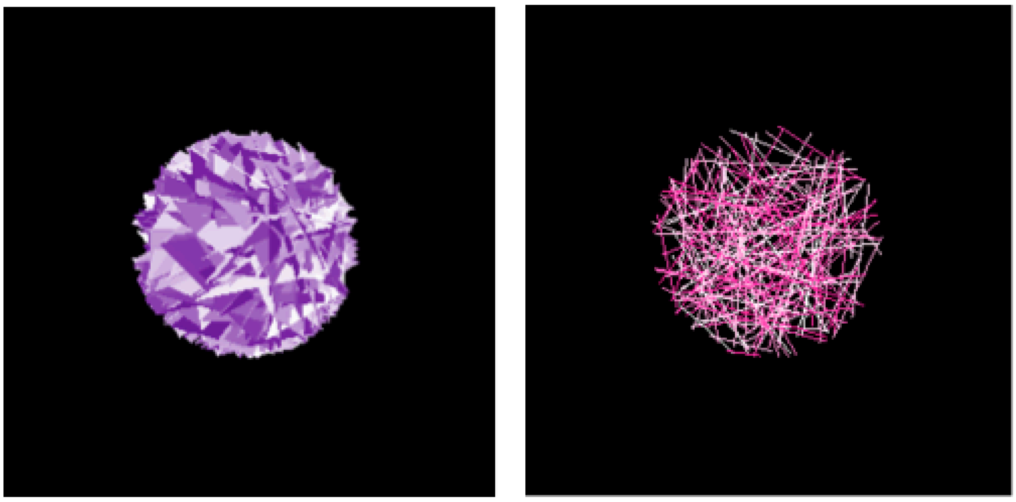


**Figure 26. Mapping Configurator: visual objects effects.**

As shown in Figure 25, one particular string´s visual parameters can be mapped to sound parameters in different ways:

- The colour for the notes could be mapped to the range of visible colours (first prototype approach); to the colours of a spectrogram (based on the energy of each note, ranging from blue/low-energy to red/high-energy); manually select the colours of the notes for a particular string; or randomly select a colour every time an onset is detected in a particular string.
- Notes location could be manually settled (X,Y and Z-axis) for each string, or automatically generated by mapping the Y-axis to the pitch height (first prototype approach) and the Z-axis to the energy of the note, including threshold parameters to configure the translation of the objects. Each axis could be independently configured, so that the Y-axis could be manually settled and the Z-axis automatically.
- The size of the objects could be mapped to the notes' energy (first prototype approach), or manually settled for the notes of each string.
- Objects could be manually rotated, or automatically by setting a speed.
- Additionally, note names can be switched on/off. This information is presented over the object representing the note, as it was the case of the First Prototype in which the note name appeared inside the circle representing the note. Additional information such as the octave number and chord name can also be displayed

Although being basic, these characteristics' configuration leads to a large flexibility when designing the visualizations. The following Figures show some examples of different visualization configurations.

As a note, in Figure 29 one can see the visualization approach of the First Prototype. In the Mapping Configurator, this visualization approach can be achieved by a particular parameter configuration, which means that the First Prototype visualization is comprised in the Second Prototype´s Mapping Configurator GUI.

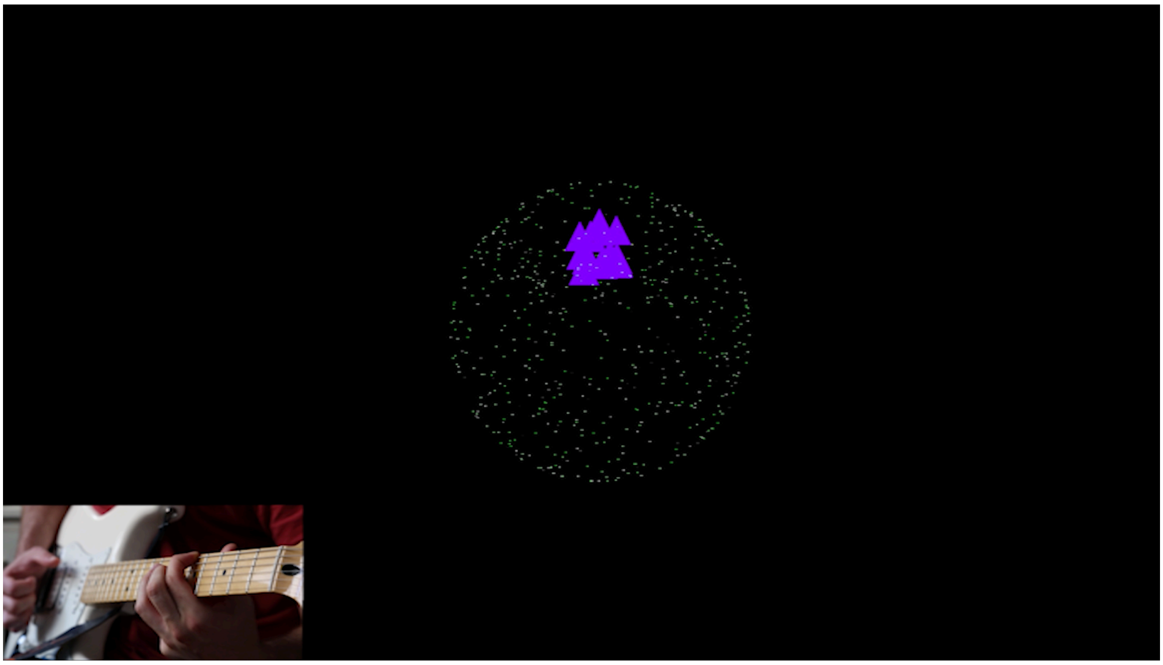


**Figure 27. Second Prototype visualization example I.**

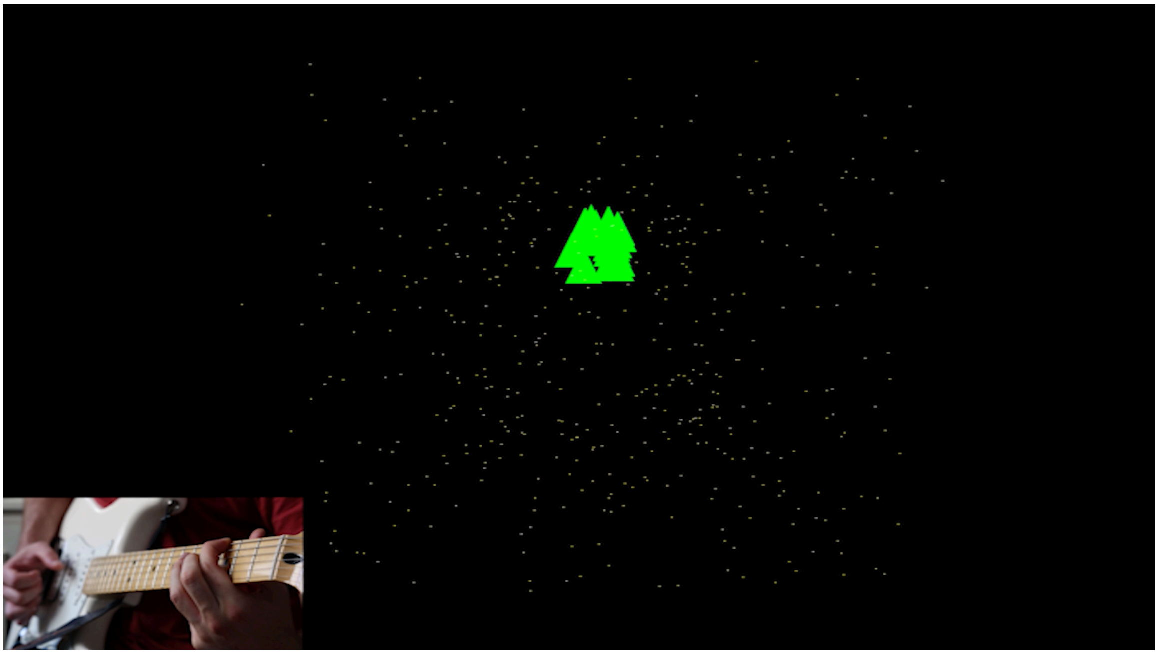


**Figure 28. Second Prototype visualization example II.**

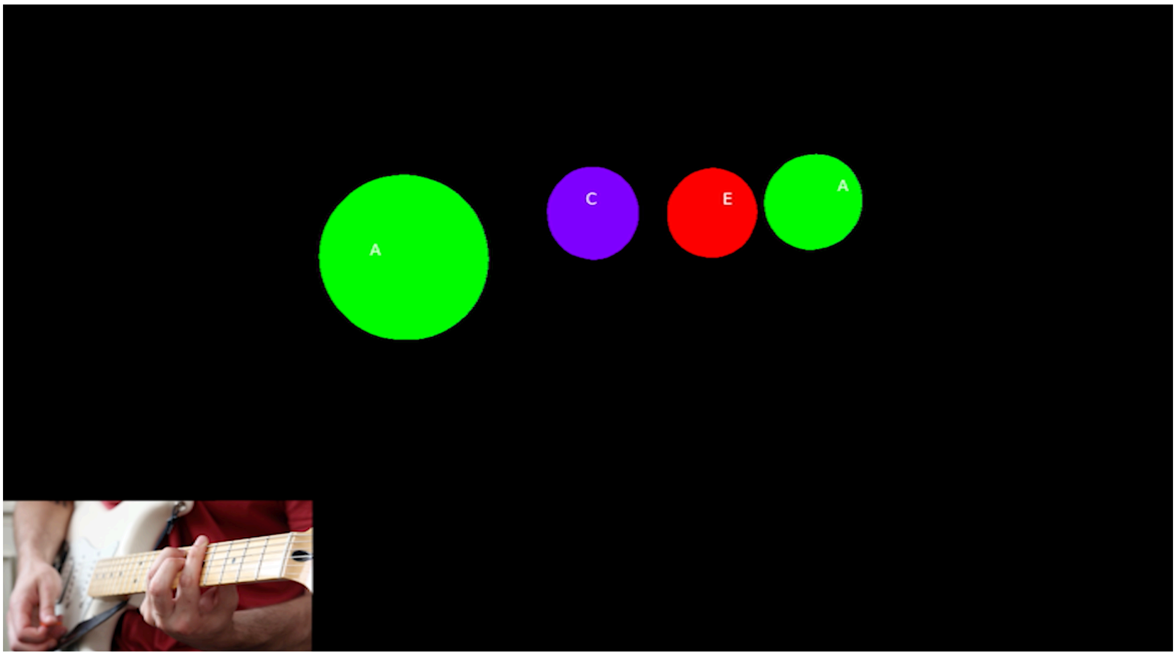



**Figure 29. Second Prototype visualization example III.**

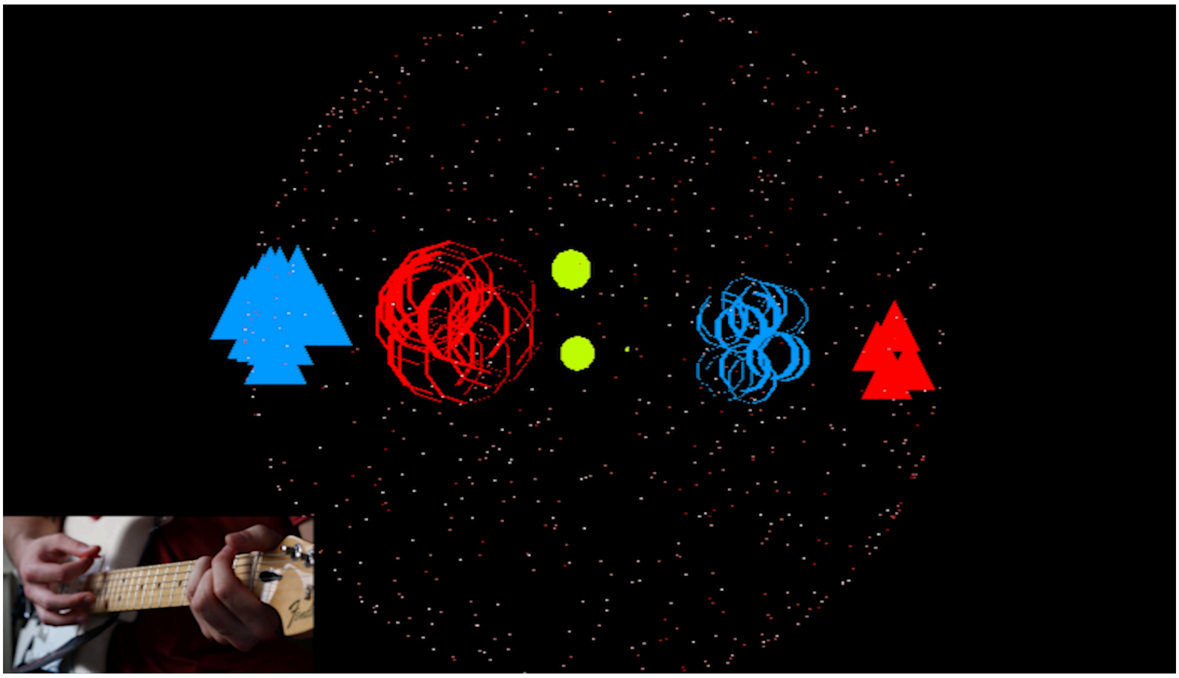


**Figure 30. Second Prototype visualization example IV.**

# 4 EVALUATION

In this section the methods selected for evaluating the prototypes are described, as well as the results and feedback we obtained from the users.

## 4.1 First Prototype Evaluation

The evaluation described here corresponds to the first prototype. It was carried out for the elaboration of the paper (Angulo et al., 2016) that summarized the work done for the first prototype.

### a) Experiments

An evaluation was prepared based on some basic guitar "riffs/phrases" visualizations. We focused on four different guitar phrases: two different chord progressions, a melody, an arpeggio, and a solo. The phrases were played in the same key, in order to produce similar visualizations (same colours, localisation of notes, etc.).

We proposed three different experiments to the users. In the first one, one of the two different chord progression recordings (Figure 31 and Figure 32) was presented to the user, and then the visualizations of the two chord progressions were shown in silence. The user had to choose the visualization that matched the audio recording.

The second experiment was the opposite, given one visualization (presented in silence), the user had to select from two recordings the one that matched that visualization, in this case, using fragments of the solo and melody phrases. In addition, the user was asked to indicate the complexity he/she found when doing the first two experiments.

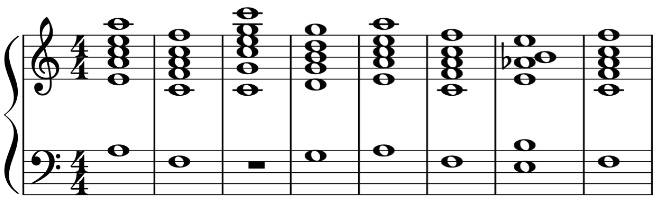


**Figure 31. Chord Progression I score.**

**Figure 32. Chord Progression II score.**

**Figure 33. Melody Score.**


**Figure 34. Arpeggio Score.**

The last test consisted of listening to all the phrases (sorted, presented as a song) together with their corresponding visualizations, and afterwards, answering some questions to rate the system.

The questions evaluated the system in terms of:

. mapping quality and meaningfulness,
. expressiveness, subjectively evaluated by the user considering if the visualizations led to a stronger experience of music (multimodal perception),
. interest, if the system was considered interesting/promising by the user
. utility, in which context would a system like this one be used by the user.

The answers consisted of a score from 1 to 5 to express agreement, disagreement or neutrality, in addition to a text box in which the users could write their opinion, suggestions, or ideas for improvements.

### b) Results

The experiment was conducted with 20 participants whose ages ranged from 21 to 55. They had different backgrounds and musical training. Besides, their musical taste was varied, as well as the frequency with which they went to concerts and listened to music. Table 1 shows the summary of the results of the experiments.

| | Correct answers | Difficulty (1-5) |
|---|---|---|
| Test 1 | 80% | 2.9 |
| Test 2 | 75% | 3.2 |

**Table 1. Experiments Results (First Prototype).**

80% of the users were able to identify the correct answer to the first experiment, with a difficulty of 2.9 (the mean of the 1 to 5 range, where 1 was easy and 5 difficult); and 75% of the users answered correctly to the second experiment. In particular, participants with musical education and/or guitar players found the task easy, were able to distinguish between the three visualizations, and even imagine how the music would sound before listening to it.

| | Score (1-5) |
|---|---|
| Mapping quality/meaningfulness | 4.3 |
| Expressiveness | 4.2 |
| Interest | 4.8 |

**Table 2. System Valoration (First Prototype).**

Table 2 shows the users valoration of the system in term of mapping quality and meaningfulness, expressiveness and interest. The score ranges from 1 to 5, in which 1 means disagreement and 5 is strong agreement. Several comments were made about the mappings. Most users found intuitive the proposed connections between the sound and visual domains, but many of them argued about the use of colour to identify notes. Also, most of the users liked the experience of simultaneous music and visuals, but some of them said the visualizations were very basic, and suggested that developing more "artistic" visualizations would work better and transmit more sensations.

All the users found the system very interesting, and suggested different contexts in which it could be used. Most of them proposed using the system in live music performances and concerts, to reinforce the emotions a particular music piece tries to evoke in the listeners; some participants suggested that the system could be used as a didactic tool to help people learn guitar playing, and musical concepts in general.

Moreover, some participants said “as a tool to emphasize sensorial experiences for infants in primary education and give support in art classes”, or even “for helping disabled people (i.e. people with hearing problems) to perceive and experience music”.

## 4.2 Second Prototype Evaluation

In the case of the Second Prototype, the evaluation concerned to the Graphical User Interface to configure the mapping between sound and vision, i.e. Mapping Configurator.

### a) Experiments

Two experiments were proposed to evaluate the Mapping Configurator GUI. In the first one, a short guitar piece (riff, phrase) audio was looped and presented to the user, accompanied by a pre-configured visualization. The aim of the user was to try to understand the musical features through the proposed visualization, and then configure the parameters of the GUI to make the visualizations more meaningful to her/his understanding. At this point, the user had to rate the flexibility the GUI offers when making changes in the visualization parameters, and its capability to adapt to the user´s intention.

In the second experiment, the user had to think of a piece and play it with the guitar (in the case the user knew guitar playing, otherwise she/he could select one between a set of guitar music audios proposed for the experiment). After listening carefully to the musical piece, the user had to imagine how would the piece look like, taking into an account the feelings and emotions it evoked. Then, the aim of the user was to try to recreate the imagined visualization through the Mapping Configurator GUI. Finally, the user had to evaluate the flexibility and capability of the system to configure the mappings, the expressiveness the controls of the GUI offers for an artistic visualization generation, and the intuitiveness of controlling the GUI.

### b) Results

In this case, the experiments were carried out with 12 users who evaluated the Second Prototype´s Mapping Configurator. The summary of the results is reflected in the following tables. The score given by the user for rating the system´s characteristics ranges from 1 to 5, being 1 poor or disagreement and 5 rich or agreement.

| Experiment 1 | Rating (1 – 5) |
|---|---|
| Mapping Configuration Flexibility | 4.2 |

**Table 3. First experiment results (Second Prototype).**

| Experiment 2 | Rating (1 – 5) |
|---|---|
| Mapping Configuration Flexibility | 3.6 |
| Mapping Expressiveness | 3.9 |
| GUI Intuitiveness | 4.1 |

**Table 4. Second experiment results (Second Prototype).**

As the tables show, the flexibility or capability of the Mapping Configurator GUI for customizing the mappings between music and visual domains results pretty high, with a score of 3.9 including both experiments. However, users considered that the system performed better when configuring a pre-established visualization´s mappings (Experiment 1), rather than the case of recreating an imagined visualization from scratch (Experiment 2). Probably, the reason for this is a lack of freedom in the visualization configuration, which remains limited in comparison to the visualization a user could imagine.

In terms of expressiveness, most of the users liked the mapping configuration possibilities, and furthermore, some of them who participated in the evaluation of the First Prototype considered an enhancement and enrichment of the system capability in the interconnection of sound and vision.

Regarding the GUI´s intuitiveness, most of the users found no difficulties when controlling and configuring the parameters, and manifested their liking and the easiness of controlling the GUI during guitar playing.

To add some comments, we found that users with musical education performed better, as they rapidly understood the involved musical features and thought of different ways to visually represent them. Guitar players probably were a step ahead of the rest of the users as they have a deep understanding of guitar, but after some minutes experimenting with the GUI, most of the users were able to use it to their will.

Finally, as improvements for the system, developing more mapping possibilities highlighted over others. Also, some users missed the interaction between sound sources, in the sense of interconnecting the behaviour of two or more strings' objects.

# 5 CONCLUSIONS

Music is one of the most powerful art-expressions, and vital part of almost every known human culture along history. Technological advancements over the last decades have brought us the opportunity to explore it from new perspectives and approaches. Nowadays, many researchers focus their work on representing music visually, by accurately transcribing musical events, but also focusing on reflecting the sensations and emotions it evokes as an artistic expression.

Our interest resides in guitar music representation to offer, through the system described in this project, a visual tool for experiencing guitar performances. For this, we focused on several perspectives, which characterise on the intention of communicating information to the user, i.e. objective visualizations that gather clear accurate information about musical events so that the user can learn from them, and subjective visualizations that create an aesthetic experience for the user by artistically representing musical events.

Once the system was developed, we carried out experiments with people from very different backgrounds. Throughout these experiments we noticed that the idea of a system for visualising music rapidly catched the users´ attention, as the multimodal experience of music and synesthetic works. Although many users of the experiments considered themselves as not musicians, some of them tried to play guitar at some point of their lives. This was very interesting for our system evaluation, as most of them were able to understand how a guitar "works", and hence understand what was our goal with the visualizations.

As the results of the experiments showed, people found the system interesting and promising in many different use contexts. These, in general terms, coincided with the division we posed for the visualization approaches, i.e. a didactic tool for learning guitar playing (*objective* visualization approach), or for artistic applications, such as live concerts' light and visual show (*subjective* visualization approach).

The developed system responds to various characteristics we found important during the design process. First of all, the system uses monophonic sound transcription techniques, which keeps the signal processing task simple. For this, a hexaphonic guitar is needed, another task of the project for which we propose two different solutions, starting from a conventional guitar. In the Second Prototype, we based the visualisations on two perspectives: the Guitar Fretboard (*objective* visualization approach) aims to be used as a didactic tool as it represents accurate musical information; and the Mapping Configurator (*subjective* visualization approach), which permits the real-time mapping configuration focused on the detailed visualization of guitar music.

For future work, more possibilities will be developed for the visualisation generation, widening the mapping configurations, regarding the use of more features in both musical and visual domains. Also, many users of the experiments showed their interest about adapting the system to conventional guitars, avoiding hexaphonic guitar transcription. This change supposes the study of different signal processing techniques for the feature extraction task, as using conventional guitars means dealing with polyphony. However, this could be a very interesting approach, so it will remain as a possible direction for the continuation of this work.

# 6 CONTRIBUTIONS

During the development of this project, we wrote a paper describing our work on Hexaphonic Guitar Transcription and Visualization for the TENOR 2016 conference, discussing music notation and representation technologies, and hosted by Anglia Ruskin University in Cambridge (UK). The paper detailed the design and development of the First Prototype of this project, as well as the evaluation we performed and the results we obtained from it. This paper can be found in the proceedings of the conference, following the link attached to the reference of the document.

Angulo I, Giraldo S, Ramirez R. (2016). "Hexaphonic Guitar Transcription and Visualisation". TENOR 2016, International Conference on Technologies for Music Notation and Representation (pp. 187 – 192), Anglia Ruskin University, Cambridge (United Kingdom), 2016.

Also, some of the material and resources generated in this work are available for future research, in an Online Repository dedicated to the project (see Annex).

# 7 ANNEX

## 7.1 Online Repository

An Online Repository has been created for sharing some of this project´s resources, so that researchers and every person interested in the topic can use this information for future work.

| |
|---|
| Link to Online Repository:<br><br>https://github.com/inigoao/HexGuitar.git |

## 7.2 First Prototype Evaluation

**EVALUATION** SMC Thesis, UPF Nov2015. Iñigo Angulo. Supervisors.: Rafael Ramírez, Sergio Giraldo.

Name:
Age:
Occupation:
What kind of music you listen to? (you can name genres, bands… whatever cames to your mind):

How often do you go to a concert?
☐ *less* ☐ *once per 6 months* ☐ *once per 3 months* ☐ *once per month* ☐ *once per 1 or 2 weeks* ☐ *more*

Do you play guitar?

Do you play any musical instrument or are involved in music production?

Do you have a music education, studied music?

Do you have any experience in music visualization, VJing, etc?

**1.** Given one audio, which the user will listen before seeing any visualization, select between 2 different visualizations (Visualization 1 & 2) the one that corresponds..

*[listen to the audio]*

*[display both visualizations]*

*Select the visualization that corresponds to the audio:*

☐*1* ☐ *2*

Please valorate the difficulty of answering the previous question:

☐ *very easy* ☐ *easy* ☐ *medium* ☐ *difficult* ☐ *very difficult*

**2.** Given one visualization, select the audio that corresponds to it.

*[display the visualization]*

*[listen to the audios]*

*Select the audio that corresponds to the visualization:*

☐*1* ☐ *2*

Please valorate the difficulty of answering the previous question:

☐ *very easy* ☐ *easy* ☐ *medium* ☐ *difficult* ☐ *very difficult*

**3.** Now you are going to see a composition of audio and visualizations. Watch the video in which music is presented with it´s visualization (simultaneously) and answer the next questions:

*[display the video]*

- <u>Visualization Synthesis</u>: valorate (from 1 to 5) the quality of the visualizations in relation with the music.

- Quality of the mappings between music and visuals. *Mapping refers to joining a musical feature with a visual one. For example, in these visualizations we find*:

| |
|---|
| Mappings into a 2D space<br><br>- Notes are represented with Circles<br>- Location of the notes:<br>» Strings --> x-axis (vertical "invisible" lines)<br>» Pitch height --> y-axis<br>- Size --> Loudness<br>- Fret (fretboard frequency map) --> Notes Names<br>- * Colours to notes --> Aesthetic (subjectivity) |

Now valorate, in terms of quality and "meaningfulness", the mappings of the visualizations you just saw.

☐ *1* ☐ *2* ☐ *3* ☐ *4* ☐ *5*

Now please write your opinion, sugestions and improvements you will apply to the visualizations:

| |
|---|
| |

- <u>Expressiveness</u>: *there exist theories in the Music Visualization field which defend that simultaneous multimodal (sight and hearing) stimulus result in an "added value"/stronger perception.*
Valorate your experience (in terms of stronger experience of perceiving music when visualizations are simultaneously displayed) of listening to the music while having a visual representation of it:

☐ *1* ☐ *2* ☐ *3* ☐ *4* ☐ *5*

- Interest: now valorate the system (and the concept behind it --> "Music Visualization"), as how interesting you find it, and musically promising as a didactic/artistic tool.

*□ 1  □ 2  □ 3  □ 4  □ 5*

- Utility and use context: please think and write in which context would you use a system like this.

- Finally, please write any comments, suggestion, idea for helping us to improve the system.

THAT'S ALL, THANK YOU!!

# 7.3 Second Prototype Evaluation

**EVALUATION** SMC Thesis, UPF 2016. Iñigo Angulo. Supervisors.: Rafael Ramírez, Sergio Giraldo.

Name:
Age:
Occupation:

Do you play guitar?

Do you have a music education, studied music?

Do you have any experience in music visualization, VJing, etc?

Through this experiment you are going to test a music visualization system for guitar music. This system processes the audio signals of the guitar to extract musical features to characterise the sounds. Some of these, such as the notes fundamental frequencies, onsests and loudness, are normally used in music transcription tasks. However, the approach presented in this experiment aims towards the creation of artistic visualization of music, mapping musical features and visual elements.

Following the idea presented before, a software called Mapping Configurator have been created to offer the user the possibility of configuring the mapings between musical and visual domains, through a Graphical User Interface that works in real-time and can be manipulated while the music is being played.

Now, you are going to evaluate this GUI through two different experiments.

1. You are going to hear a pre-recorded guitar performance audio, together with a pre-configurated mapping for sound and visuals. The mappings are:

Mappings into a 2D space

- Notes are represented with Circles
- Location of the notes:
  - » Strings --> x-axis (vertical "invisible" lines)
  - » Pitch height --> y-axis
- Size --> Loudness
- Fret (fretboard frequency map) --> Notes Names
- * Colours to notes --> Aesthetic (subjectivity)

Think about how meaningful are these mappings to you. Through the controls offered in the GUI, try to change the paremeters to make the visualisation of music more meaningful. Once you are happy with the configuration, please valorate the system in terms of the flexibility and the capability it had to represent musical features as you understand or imagine them.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

2. Now, think of a guitar phrase or riff and play it. (If you don´t play guitar do not worry, select one of the phrases offered for the experiment). Listen carefully to it, and try to imagine how this music would look like, taking into account the emotions and feelings it evoques to you. Now, try to create the visualization with the tools offered in the GUI. Valorate the system in terms of visualisation creation flexibility/capability.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

Valorate the system in terms of the expressiveness it permits to reach through the visual tools it offer.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

Valorate the system in terms of the intuitiveness when controlong the GUI.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

- Finally, please write you thoughts about the system, and any comments, suggestion or idea for helping us to improve it.

| |
|---|
| |

THAT'S ALL, THANK YOU!!

# 8 BIBLIOGRAPHY


Alves, B. (2005). Digital Harmony of Sound and Light. *Computer Music Journal*, *29*(4), 45–54. doi:10.1162/014892605775179982

Angulo, I., Giraldo, S., & Ramirez, R. (2016). HEXAPHONIC GUITAR TRANSCRIPTION AND VISUALIZATION. In *TENOR 2016, International Conference on Technologies for Music Notation and Representation.* (pp. 187 – 192). Cambridge: TENOR 2016. Retrieved from http://tenor2016.tenor-conference.org/TENOR2016-Proceedings.pdf

Argenti, F., Nesi, P., & Pantaleo, G. (2011). Automatic Music Transcription: From Monophonic to Polyphonic. *Musical Robots and Interactive Multimodel Systems*, 27–46.

Bain, M. N. (2008). *Real Time Music Visualization: A Study In The Visual Extension Of Music*. The Ohio State University.

Bergstrom, I. (2011). *Soma: live performance where congruent musical, visual, and proprioceptive stimuli fuse to form a combined aesthetic narrative*. UCL. Retrieved from http://discovery.ucl.ac.uk/1310143/

Bogdanov, D., Wack, N., Gómez, E., Gulati, S., Herrera, P., Mayor, O., … Serra, X. (2013). *ESSENTIA: an open-source library for sound and music analysis*. doi:10.1145/2502081.2502229

Evans, B. (2005). Foundations of a Visual Music. *Computer Music Journal*, *29*, 11–24. doi:10.1162/014892605775179955

Fiss, X., & Kwasinski, A. (2011). AUTOMATIC REAL-TIME ELECTRIC GUITAR AUDIO TRANSCRIPTION Xander Fiss , Andres Kwasinski Department of Computer Engineering , Rochester Institute of Technology , Rochester , NY, 373–376.

Guitar Hero Official Site. (2005). Retrieved November 1, 2015, from https://www.guitarhero.com/

Ivry, R. (2000). A Review of Synesthesia. *University of California, Berkeley*. Retrieved from http://jheer.org/publications/2000-Synesthesia-Psych127.pdf

Katja Vetter. (2012). helmholtz finds the pitch. Retrieved from http://www.katjaas.nl/helmholtz/helmholtz.html

Klapuri, A. P. (2004). Automatic Music Transcription as We Know it Today. *Journal of New Music Research*, *33*(3), 269–282. doi:10.1080/0929821042000317840

Knesebeck, a Von Dem, & Zölzer, U. (2010). Comparison of Pitch Trackers for Real-Time Guitar Effects. ... *Int. Conf. on Digital Audio Effects ( ...*, 1–4. Retrieved from http://dafx10.iem.at/proceedings/papers/VonDemKnesebeckZoelzer_DAFx10_P102.pdf

Magic Music Visuals: VJ Software, Music Visualizer & Beyond. (n.d.). Retrieved June 14, 2015, from https://magicmusicvisuals.com/

O'Grady, P. D., & Rickard, S. T. (2009). Automatic hexaphonic guitar transcription using non-negative constraints. *IET Irish Signals and Systems Conference (ISSC 2009)*, 22–22. doi:10.1049/cp.2009.1699

Pramerdorfer, C. (2011). *An Introduction to Processing and Music Visualization*.

Processing. (2001). Retrieved June 22, 2015, from https://processing.org/

Puredata.info. (2016). Pure Data Community Site. Retrieved from https://puredata.info/

Rocksmith. (2011). Retrieved November 1, 2015, from http://rocksmith.ubi.com/rocksmith/es-es/home/

Siedenburg, K. (2009). An Exploration of Real-Time Visualizations of Musical Timbre. *Welcome to the 3 Rd International Workshop on Learning Semantics of Audio Signals*, 17. Retrieved from http://wwwiti.cs.uni-magdeburg.de/~stober/publ/lsas2009proceedings.pdf#page=23

Unfretted - Fretless Guitar Resource. (n.d.). Retrieved June 22, 2015, from http://unfretted.net/loader.php?LINK=/profs/breakout

Yousician. (2010). Retrieved November 1, 2015, from https://get.yousician.com/