Music to Color Barcode

Zoë James, Molly Robins, Elizabeth Shaher

Department of Electrical and Computer Engineering
University of Rochester

1. Abstract

When consuming music we rely solely on our ears to help us understand different characteristics of a piece. These characteristics, such as loudness, pitch, and timbre, are important distinguishing features that can define the music we listen to. We believe that we could use other senses to understand these characteristics and how they relate to a piece as a whole. In this paper, we will explore representing music and its components with color over time as colored barcodes. We will also discuss some of the applications for these barcodes.

2. Introduction

Going into this project, we knew we wanted to create a visual representation of music or sound in some form. Some humans are born with synesthesia and certain types of synesthesia allows them to hear sounds and experience them as color. We wanted to create something similar, but we knew that everyone would experience sound differently and so we could not survey people for their input on what certain sounds look like. We came across a TED talk by Neil Harbisson who is color blind and sees the world through frequencies. He created a piece of hardware that he wore that would see colors and translate it to a different frequencies. We sought out to do the opposite, by creating a project that read music and sounds and translated it to different colors over time. We decided that printing a color for every frame over time as a single static picture would best represent our data. The printing of these colored frames produced a barcode.

3. Background and Research

3.1 Relation Between Color and sound

From our research, we found that there is no direct relationship between color and sound. They are both frequency based, and color is created through additive wave synthesis, but there is no corresponding color that can be associated with a specific sound. There are certain people with a condition called Audio-Visual Synesthesia that do associate colors and sounds, however, the condition varies from person to person, so psychological research could not be used as a direct relation either.

With no direct mathematical or psychological relationship between color and sound, we had to design our own. We decided to create a relationship between parameters of our audio and RGB colors. RGB color code is a ratio accepted by computers to create all possible digital colors. The colors red, green, and blue are all given a value between 0 and 255; the ratio of those values corresponds to a specific color. The code (0,0,0) is associated with black, while the code (255, 255, 255) is associated with white. By implementing RGB colors, instead of solely frequency-based color recreation, we could account for three different parameters of an audio file instead of just one.

3.2 Key Components of Musical Analysis

The components that were chosen for musical analysis were inspired by components used for genre classification. In short, genre classification uses timbre, rhythmic content, and pitch content to describe a piece and its content. Timbre was classified using several values that can categorize how the spectrum of a piece looks. The spectral centroid is a calculated value that can describe brightness, or where most of the frequencies of a piece lie in the spectrum. Other important timbral qualities can be described with rolloff (describes “spectral shape”), flux (describes how the timbre changes over time), “time domain zero crossings” (describes levels of noise), and energy (describes quietness). Rhythmic content was classified with a series of analyses that calculated beats per minute and several other related values.

Pitch content was classified with a series of pitch detection algorithms, followed by related analyses.[5]

From this research, we were able to decide on three components that could be used to describe a piece, varied noticeably over time, as well as were relevant to calculate. Firstly, we chose loudness, to mirror how energy was included in genre classification. This is a very important component to how sound is perceived, and can separate a piece's sections. Secondly, we chose the spectral centroid, to represent timbre and brightness. Brightness is a very noticeable component within a piece, and can vary quite a bit. It is also an easy way to characterize the spectrum with a single value. Lastly, to model pitch content, we chose the loudest frequency as our third component. The loudest frequency would be largely affected by any prominent instruments present in the piece, and can also change interestingly over time.

4. Description of the Project

4.1 Methods

This project was written in MATLAB, as the program provides many tools for audio analysis and visualization. The code is comprised of a single main script that calls two functions: one that calculates the key components, and another that translates those key components into an RGB color code. Each of these functions operate on a single frame. The main script itself is where the sound file is input, and the barcode is generated.

Before analysis, the input is buffered into frames with a frame length of 512 samples, using a rectangular window and a hop size of zero samples. This frame window is short enough to catch the many variations within a single piece, while still maintaining enough length to create a number of frequency bins used in analysis. The rectangular window was chosen such that the content in the frame is not changed or improperly weighted for analysis. The zero hop size was chosen as no overlap between frames is needed, and this allows the barcode to more closely represent the content of the input.

After the buffering process, the input is processed frame-by-frame through the spectral analysis function followed by the sound-to-color converting function. The output of the sound-to-color conversion is then plotted as a stem plot with a wide line thickness, without the stem heads.

4.2 Key Component Analysis

For both the centroid and the loudest frequency calculations, the spectrum was calculated using a Discrete Fourier Transform through the Fast Fourier Transform function in MATLAB. The single-sided magnitude spectrum was used, calculated using a method described in documentation. We also decided not to perform any normalization on the spectrum or the loudness, as it could cause unwanted modifications to the parameters we are using to create the visualization.

The loudness of the frame is stored in the output as a value in decibels. This value is calculated by simply averaging the amplitude values at each sample in the frame, and converting to decibels using the equation described below.

Equation 1: Linear to Decibel Conversion

$$y_{dB} = 20\log_{10}(x_{lin})$$

To calculate the centroid, we used the formula as described in Tzanetakis and Cook's genre classification research, included below.[5] This formula yields a ratio that describes the relationship between the centroid bin and the total number of bins.[2] Using this ratio, the centroid bin was calculated by multiplying by the total number of bins and rounding to the nearest integer below the value. Using methods described in MATLAB documentation, the frequency bin was converted to a frequency in Hertz. This frequency is what is stored in the output.

Equation 2: Spectral Centroid Formula

$$C_t = \frac{\sum_{n=1}^{N} M_t[n] * n}{\sum_{n=1}^{N} M_t[n]}$$

The maximum frequency was found by finding the frequency bin at which the maximum magnitude occurs. This frequency bin was also saved in the output as a value in Hertz.

4.3 Sound to Color Conversion

After coming up with the functions to extract the different characteristics from the song, we then had to match them to the three values that make up the RGB color code. The red color was associated with Loudness. The value was calculated using the formula below.

Equation 3: Decibel to Red Color Value Conversion

$$Red = 255 - (3 \cdot |dB|)$$

Our reason for choosing to multiply our decibel level by 3 is that the most quiet sections of music are about -80dB. When there is no scaling on the decibel value, the barcodes become overwhelmingly red.

The green color was associated with the brightness, or the centroid, of the frame within the song. The blue color is associated with the loudest frequency of the frame. Since green and blue are both based on frequency, they can be calculated the same way. To associate a value between 0 and 255 with a frequency that can range anywhere from 0 to 20,000 we took the common frequency bands used for mixing into account. There are 7 frequency bands commonly used in mixing. Below is a table of the common mixing frequency bands.

Table 1: Common Frequency Bands in Mixing

Band Number	Frequency Range (Hz)	Band Name
1	≤ 50	Sub-Bass
2	60 → 250	Bass
3	250 → 500	Low-Mid
4	500 → 2000	Mid
5	2000 → 4000	Upper-Mid
6	4000 → 6000	Presence
7	6000 ≤	Brilliance

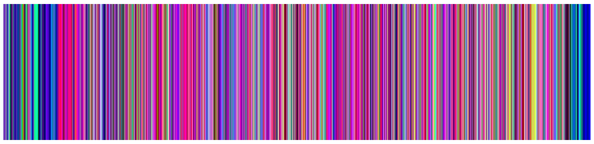
To associate the green and blue values with the frequencies, we multiplied the band number of the input frequency by 20 to account for the specific band. We then modulated the frequency over 135 and added it to the multiple of 20. This allowed us to account for differences between frequencies within the same band.

5. Results and Discussion

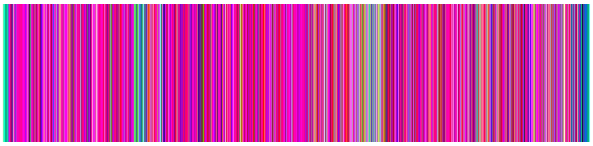
5.1 Music to Color Conversion Barcodes

Figure 1, shown below, displays Zoë's, Molly's, and Elizabeth's reference mixes in order from top to bottom. These are songs that we know very well and often use them to understand the frequency responses of spaces and electronics.

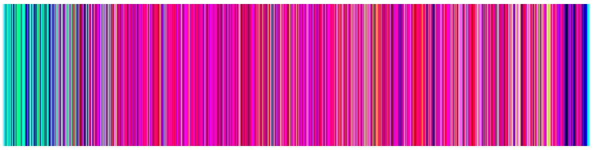
Figure 1: Barcodes of Reference Mixes



Michael Jackson, "Beat It"



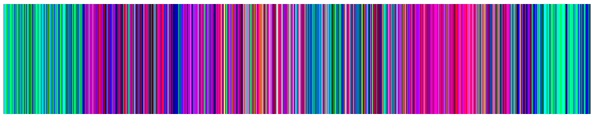
Tame Impala, "Let it Happen"



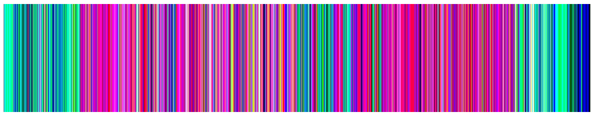
My Chemical Romance, "Welcome to the Black Parade"

To show how the barcodes are individual to the song, we created barcodes for two different versions of the same song, "Bohemian Rhapsody". The first barcode in the figure below was made using original song, and the second barcode in the figure was created using a cover of the song that emulated the style, but was performed by a different artist and produced by a different team.

Figure 2: “Bohemian Rhapsody” Comparison



Queen, “Bohemian Rhapsody”



Panic! At The Disco, “Bohemian Rhapsody”

The barcodes for these two versions of the same song are visually different, and can be properly matched within our database. However, it is also clear that these songs are very similar. The frequencies present and the volume of each section of the song are relatively similar. They look like the same song, but can be differentiated, the same way that they sound like the same song but can easily be differentiated.

5.2 Music Identification Through a Database

One of the best applications of our visualization software is the ability to use it as a way to represent and recall data. Since it is not possible to perfectly recreate the song, or any piece of sound from the color barcode, we can use them as representations of the audio. Each barcode is different enough that they can be distinguished from one another using a program that can read each color individually. We created a small database of 54 songs with their associated barcode. Our code can read and accurately match the barcode to the correct song, including matching specific versions of the same piece of music.

Our database currently works by matching an input matrix with one of the matrices stored from previous musical outputs, meaning a new song can not be identified, the barcode must exist in the database beforehand. When a song does exist in the database, the identification program will output the title and artist of the song, then play a 30 second clip of the piece of music. The drawback to using this method is that the database of songs and barcodes must be manually updated, and there must be a very rigid naming convention, as barcode matrices and songs are stored separately. Another possible way to create this database would be using object oriented coding. However, even then a very strict organization convention would be extremely important.

6. Future Work & Applications

A much larger scale database would need to be created and maintained in order to make this way of representing data usable for a larger audience.

A possible way to make this accessible to a larger audience would be to create a web based version of this code, where users could input their own audio. One of the draws of this code, beyond being a possible way to represent data, is that it is fun. Allowing users to create their own barcodes, and adding them to a central database automatically could be a way to circumvent the creation and maintenance of a personal database.

This program could also be used to maintain and organized sound design libraries. One of the problems that comes with recording long, ambient pieces of audio is the inability to find exactly where certain events happen within the recording. Creating visualizations of ambient recordings would allow sound designers to see where changes in volume or pitch occur, making it easier to find for proper use.

With further research on the topic of genre classification, this program could be modified to allow for more than just data representation. The parameters that we chose are good for differentiating different pieces of music from each other, but using new parameters, or a new way to represent colors digitally, could allow us to use this code for genre classification. Creating a way to include rhythmic content in our visualizations, either through opacity or negative space, would allow us to use this program as a way to digitally classify genre. It would not be a perfect system, but it could be possible.

Changing our color parameters could also allow us to use this program for different types of audio. MFCCs are a better way to categorize speech. Implementing MFCCs could allow us to create a program similar to ours that could recognize speeches, or even different speakers.

7. References

1. "Audio Spectrum Explained," *Teach Me Audio*. [Online]. Available: https://www.teachmeaudio.com/mixing/techniques/audio-spectrum/. [Accessed: 30-Apr-2019].

2. "jAudio 1.0 Feature Appendix," *Spectral Centroid*. [Online]. Available: http://jaudio.sourceforge.net/jaudio10/features/spectralcentroid.html. [Accessed: 30-Apr-2019].

3. Libretexts, "Electromagnetic Radiation," *Chemistry LibreTexts*, 23-Apr-2019. [Online]. Available: https://chem.libretexts.org/Bookshelves/Physical_and_Theoretical_Chemistry_Textbook_Maps/Supplemental_Modules_(Physical_and_Theoretical_Chemistry)/Spectroscopy/Fundamentals_of_Spectroscopy/Electromagnetic_Radiation. [Accessed: 30-Apr-2019].

4. "Light: Electromagnetic waves, the electromagnetic spectrum and photons," *Khan Academy*. [Online]. Available: https://www.khanacademy.org/science/physics/light-waves/introduction-to-light-waves/a/light-and-the-electromagnetic-spectrum. [Accessed: 30-Apr-2019].

5. *Musical genre classification of audio signals - IEEE Journals & Magazine*. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/1021072. [Accessed: 30-Apr-2019].

6. "Neurowiki 2014," *Auditory-Visual Synesthesia - Neurowiki 2014*. [Online]. Available: http://neurowiki2014.wikidot.com/individual:auditory-visual-synesthesia. [Accessed: 30-Apr-2019].

7. N. Harbisson, "I listen to color," *TED*. [Online]. Available: https://www.ted.com/talks/neil_harbisson_i_listen_to_color?language=en. [Accessed: 01-May-2019].

8. "rgb2hex and hex2rgb - File Exchange - MATLAB Central," *rgb2hex and hex2rgb - File Exchange - MATLAB Central*. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/46289-rgb2hex-and-hex2rgb. [Accessed: 30-Apr-2019].

9. "Understanding Hexadecimal Colors is Simple," *Pluralsight*, 19-Jan-2015. [Online]. Available: https://www.pluralsight.com/blog/tutorials/understanding-hexadecimal-colors-simple. [Accessed: 30-Apr-2019].