Tania Christensen
DT2217

Music visualisation with p5.js

This brief report will describe the work that resulted in a webapp that can be used for music visualisation. The purpose was to obtain knowledge of how JavaScript, more specifically the p5-library, can be used for creative coding and music analysis. This was done by developing a simple app that analyses an mp3 file of the user's choice and then uses the information to create visuals that are easy on the eye.

The initial idea was to use the Spotify API and build an app that generated visuals based on song characteristics like key, bpm, genre, while also analysing amplitude, frequency etc. However, that idea had to be abandoned as there is no way of analysing a song with regards to the latter characteristics if only using the Spotify API.

In the current version of the app, the user uploads an mp3 file that is analysed using the p5-library. Using p5's built in p5.Amplitude, the amplitude of the input song is visualised through a square, which size changes according to the amplitude. I used FFT-analysis for computing the amplitude at different frequencies. I chose to visualise this by drawing *lightning* across the whole screen. I used a function for simple beat detection called PeakDetect. If a beat is detected, it is shown through a circle that slowly shrinks should there not be a beat present.

Having no experience with p5.js I decided to just play around with examples from the documentation and with ideas I got from looking at demo's on the internet. The square and the circle might not be very exciting, but given the scope of the project, I thought it was quite successful in communicating the very basics of a song.

I added the stars as I thought they were nice touch. At the moment they do not represent anything in the music.

Colors

I wanted to use colors that corresponded to the key of the song, for example a dark color palette for keys in minor and brighter for keys in major. However, as I could not get access of that information I decided to randomly assign a "key" whenever a song is played and let the app behave as initially planned. This means, should that information become accessible, it would be very simple to adjust the code. I used randomColor.js for the colors, which proved to be a very simple and elegant solution when wanting to create shapes with gradients.