Multimedia Systems: 3D Emotional Mixer

By: Mees Gelein (s1117378), Jordy van Miltenburg (s), & Martijn Wester(s)

Concept

In the first section of this paper we will be discussing the origin behind and the idea of the 3D emotional mixer. We will first discuss the definition of 3D emotion, and then continue to talk about current systems that are used in background music settings. Finally we will conclude this section by explaining how this project can be used in a novel way.

3D Emotion

The concept of 3d emotion has been around since 1980[1]. The idea is simple. According to Mehrabian (1980) we can place all possible emotions inside a 3d space. This makes it possible to represent every emotion as a 3d vector, with each of the components specifying a certain emotional axis. The axes are Pleasure, Arousal and Dominance. Pleasure is best described as the positive-negative axis. Arousal is easily understood by seeing it as energetic content. Dominance is explained as how intimidating or 'present' an emotion is.

Background Music

There are a lot of cases in entertainment where background music is a vital part of the performance. It is used so often that we are not always aware of its ubiquitous nature. Adaptive background music has long been a challenge, especially in dynamic media like video games or theatre. In these genres people have often been using layered approaches, just stacking more or less instruments in a mix depending on the wanted soundtrack, or just horizontal scoring. The latter is a fancy term for crossfading. It essentially means that one track fades into the other. Neither of these approaches can handle much emotional scope, or cover much of the emotional space described above.

Novel Usage

We want to create a system in which we can tune the desired emotion while the background music will seamlessly flow into the desired state. This would mean that you could tune the system on the fly. For example, if, in a video game, you would win, you could make the soundtrack more energetic and more positive, whilst if you were in a creepy place, you could have the soundtrack be less present, and less energetic but still negative, making the atmosphere feel 'loaded'.

---

[1] Mehrabian, A. (1980). Basic dimensions for a general psychological theory: Implications for personality, social and environmental psychology. Oelgeschlager, Gunn & Hain Inc., Boston, MA, 39-53.

Such a system could be used for example in improvisational theatre, tabletop role playing, or in escape rooms. The digital engine could also be repurposed to be used in a video game. We honestly believe that this system could be a good replacement in any situation where adaptive background music is required.

Architecture

In the second section we will be explaining the three major parts of this emotional mixer. For each of the parts we will be discussing how they were built and how the function internally. First we will discuss the hardware construction, then the software creation. Then we will end this section by talking about the music, both about composition and format.

Hardware Construction

The hardware controller, while not a novel component of this project, still was a central part of the development of this concept. From our first discussions we already agreed upon the fact that we all liked the intuitiveness of hardware controls.

During development a couple of ideas were explored. We initially wanted to go for rotary encoders, since they would be more flexible in their rotation. Additionally they would give us the option of having a button feature built in, which could be used for example to store and restore user settings. However, both due to time constraints and due to being more intuitive controls we decided to make simple potentiometers our main input.

There are five potentiometers. Each of the potentiometers is constantly being polled by a loop on the ESP8266. It then reads this input number and sends it over OSC[2] to the computer software, which interprets it further.

The ESP8266 microcontroller was chosen for its wireless performance. It has become an industry standard for small development boards which need WiFi, or bluetooth. Recently a new iteration, the ESP32, has become available. However, this was considered overkill for our needs. The ESP8266 is more than fast enough for our needs.

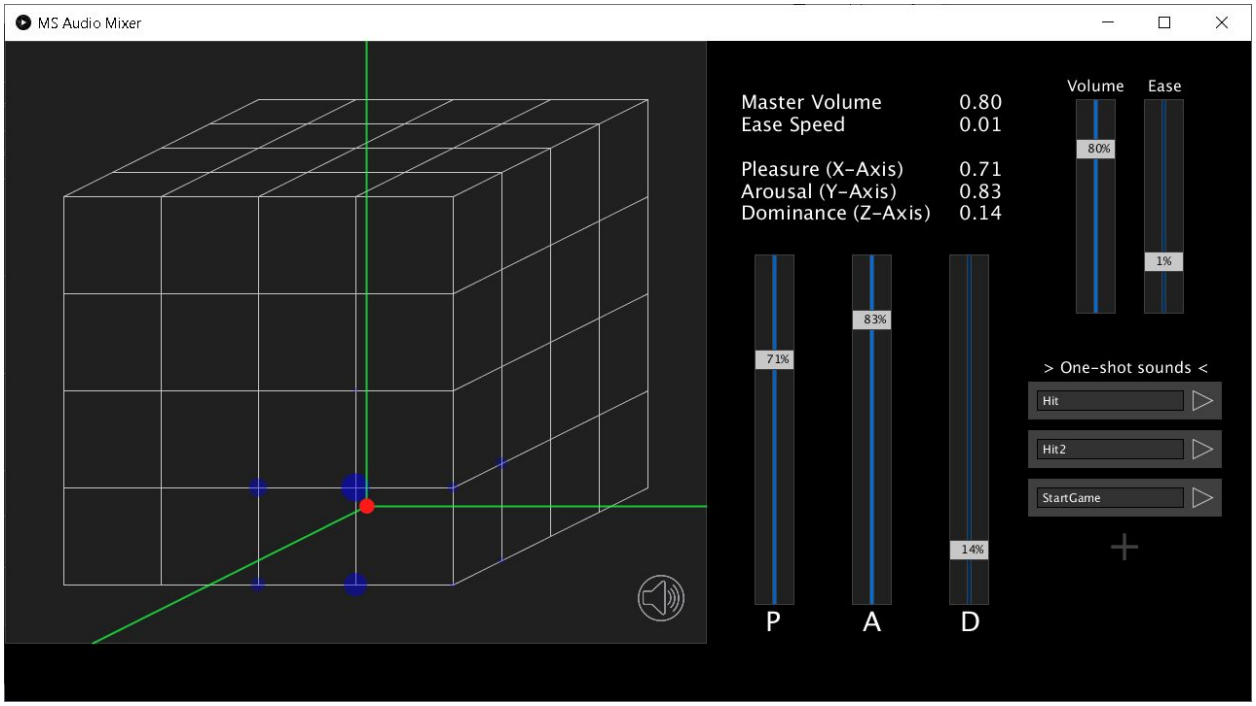
This contraption, the ESP with the five potentiometers, was then carefully mounted inside of a wooden box. The potentiometers were fitted with nice looking knobs, and a powerbank was placed inside of the project to power the controller, making it really wireless.

Software Architecture

While the true novelty of this project comes from the way all three components interact in the final product, the software is the most important part. All of the software was written in

[2] Open Sound Control is a protocol for networking sound synthesizers, computers, and other multimedia devices. (Wikipedia, checked on 13 december 2019).

Processing, a Java library. This language was chosen due to familiarity, allowing us to develop at a faster pace.



The idea behind this software is quite simple, yet a lot of tuning is necessary for the final project to work as intended. If we imagine the 3D emotional space as a cube, we can describe any point in this cube with a three dimensional vector. For example: (0, 0, 0) could be thought of as, no pleasure (i.e. negative), no arousal (i.e. low energy), and no dominance (i.e. low presence).

If we then, during loading, assign vectors to each of the sound files, we effectively position them in emotional space. We can now use Euler distance to only play the *n* closest sounds or the sounds within a radius of *r*. Most of the tuning of this program is related to how it handles the proximity of files.

It is necessary for the alignment of the music files to have all of them playing at all times. Since the ‘resolution’ of our musical grid is 5, there are 125 music files playing at all times, it’s just that most of them are (almost) muted. Only the closest files play, with the closest files to our current emotional setting playing loudest. In the end we tuned to project to play roughly four to five sounds around a position, giving us a nice mix of sounds.

Music Composition

While the engineering parts of the project each had their bumps in the road of development, the musical side of this project definitely was the most bumpy. In true creative fashion we vastly

underestimated the amount of time required to compose music that is compatible along three axes.

Within the first month we had a two dimensional demo running. This demo only included the pleasure and arousal axes, which proved to be the easiest to implement. Lulling us into a false sense of security we now thought we had plenty of time left.

The arousal axis controls how many instruments are playing and how energetically they are playing. Higher settings give you instruments that are playing more notes per minute, lower settings tend towards slower and lower instruments. The dominance axis controls the presence of the instruments. Both the volume and the intensity with which they are being played. The pleasure axis changes the tone of the music by modulating. The lowest values plays music in a minor or even diminished scale. The highest values for pleasure causes the orchestra to start playing in a major or even augmented scale[3].

The real difficulty emerged when we tried to implement the third axis. This axis caused major musical clashes, since it changed keys to convey its emotion, as discussed above. Most of the music had to be recomposed for the system to sound good.

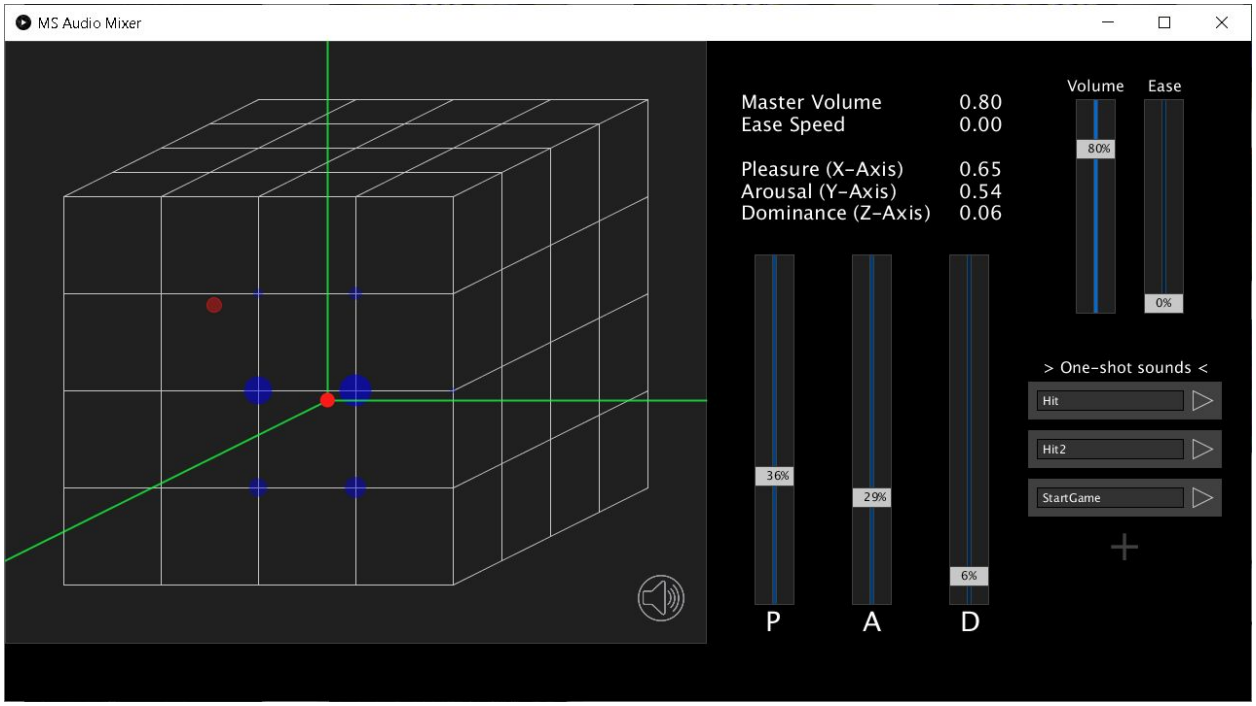
We ended up composing five different two dimensional layers, each of which where composed to fit with the layer before and after it. The extremes of the spectrum are relatively easy. Near the middle is where the real problems occur.

After a rather stressful week of deadline crunching we managed to mix 125 files to place in the emotional space. Each of the sounds is placed on one of the points in the grid by giving the filename a specific name which includes its emotional content. For example ‘mix000’ is the mix that is supposed to be placed at the origin of the 3d emotional space.

Instructions

To use this project there are a couple of prerequisites. Since we have made a hardware controller and there is unfortunately no reliable way yet to compress a physical object into a digital .zip file, we will not be able to include the hardware controller which was described in the last few pages. This was discussed with and okayed by Dr. Lew.

[3] We won’t go into too much depth about music theory here. But suffice to say that mixing these modularities in a meaningful and artistic way is a huge undertaking.



The software can be downloaded from GitHub. There are a couple of prototypes included in the system. The First Prototype folder contains, unsurprisingly, the first prototype, written by Mees Gelein. The HardwareInput folder contains the code used to turn a ESP8266 microcontroller development board (a NodeMCU) into a remote controller for a specific laptop. This code was written by Jordy van Miltenburg. All other folders contain code for different incremental upgrades and prototypes, written by Martijn Wester. The following repository is open source, licensed under MIT: https://github.com/MartijnWester1/MultimediaSystems/tree/master.

We chose to use the MIT license since it is quite a permissive license, however it does not allow other people to copy or sell our code, or major parts of it, without crediting us.

The software for the computer was written using Processing[4]. So, to run a sketch file written in Processing you would have to download the distribution appropriate for your system. The download link is: https://processing.org/download/.

When processing is installed you can open the sketch file at: /First Prototype125-v3_1shots/PAD/PAD.pde. This opens the file in the Processing IDE. Then you can run the code using the 'Run' button in the top right.

[4] Processing is a library for Java, which is aimed at Rapid Prototype Development as well as novice coders. We have chosen this environment for its ease of development and relatively good sound support.