COMP371 - Winter 2023
Adrien Tremblay - 40108982

Project Proposal - 3D OpenGL Audio Visualizer

Motivation

I am a big fan of music. I have always found visualizations for music interesting as a form of art. These visualizations used to be commonplace in music playing applications from the 90's to 2010's. So these effects are quite intuitive and simply involve using signal processing to represent the frequency distribution of the sound at any given time usually using bars. Other audio visualizer effects are more abstract. But, all audio visualizer effects have one thing in common: they involve capturing audio information, processing it to extract information, and then using this information to drive a visual effect. My goal for this project would be to create an audio visualizer program with multiple different effects that uses OpenGL to render.

Objectives

Audio Capture

The project will be a real-time application but instead of from user input, the majority of input into the application will be audio information. Ideally the application will be able to visualize whatever audio the user is currently playing on their computer. As such, the program will need to interface to and harvest information from the user's computer. For Linux systems, the A.L.S.A. audio library which is present on most Linux distributions will be used to capture live audio. While the S.F.M.L. library will be used to read audio from audio files.

Audio Processing / Analysis

After audio has been captured from the computer and is available as an audio stream, the next step is to process this audio stream to extract specific data values from the stream. This data will then be used in real time by the application to generate the visualizations. In the real world, sound occurs because of compressions and decompressions that travel through air. To record this sound, these compressions and decompressions are captured across time. The raw data represents the position the speaker head needs to move to recreate the sound at each moment in time. This data can be represented as a wave. The amplitude of this wave represents the loudness of the sound, and the frequency of this wave the pitch of the sound. But of course music consists of many different sounds playing at different frequencies. These different sounds are combined in the air to create the final sound wave. So to extract these different frequencies we need to perform signal processing. And specifically use a Fourier Transform to extract the different waves that combine to create the final sound wave. I will use the F.F.T.W library to perform my Fourier transforms.

COMP371 - Winter 2023
Adrien Tremblay - 40108982

2D Visualization - Frequency Graph / Wave

This visualization will be the classic and most intuitive visualization of audio. The audio is graphed using a series of bars. Each bar represents a small frequency range. The height of each bar then represents the amplitude of the sound at this frequency.

3D Visualization - Frequency Soundscape

This visualization will be similar to the previous one except the bars will be three-dimensional and exist in a 3d world. One possible arrangement is arranging the bars in a circle. Two circles of bars could also be rendered, one for each Left and Right audio channel.

3D Model Visualization

Another popular audio visualization is to simply render a 3D model and alter it according to the audio information. This alteration can be the position, color, texture, dimensions, etc… of the 3D Model. A library may be used to load the model.

References

- https://youtu.be/1RIA9U5oXro
- https://youtu.be/9z5HOZ4C1KY
- https://www.youtube.com/watch?v=Ibrf6LHloGc
- https://youtu.be/VXWvfrmpapI
- https://medium.com/analytics-vidhya/how-to-create-a-music-visualizer-7fad401f5a69
- http://equalarea.com/paul/alsa-audio.html
- https://youtu.be/2lZZhTf5uHY
- https://youtu.be/lfcehfutGBM