Development and Implementation of an Immersive Sound System

CS 489: Advanced Topics, Computational Audio

Smriti Sharma, Tarun Velicheti

Dr. Richard Mann

University of Waterloo

Project Progress - P1 Report

March 15, 2024

Introduction

In this project, we are exploring computational audio with a specific focus on the development of an immersive sound system. The significance of this project lies in its potential to revolutionize the way we experience sound in virtual environments and in our day-to-day lives. We aim to create a sound system to deliver high-fidelity three-dimensional sound directly to the user's headphones.

The problem we set out to solve originates from the limitations of current sound systems, which often rely on complex and expensive multi-speaker setups to deliver immersive audio. These systems, while powerful, are not practical for personal use and lack the means to provide a truly enveloping auditory experience that can be personalized for individual users. Our proposed solution is an intelligent platform that deciphers and models sound waves, enabling us to transform standard audio files into rich and dynamic auditory landscapes that we often refer to as "8D sound."

We are mindful of the scope of our project, including its limitations and boundaries. Technical challenges such as file format inconsistencies and library incompatibilities have been identified and will be addressed in subsequent phases. Additionally, while our initial work has not integrated machine learning models due to these challenges, we aim to incorporate these models as part of our future work to enhance the system's capabilities.

Literature Review

Through the early phases of the project development, we focused a lot on research before we started any coding. Our research encompasses the intersection of sound technology, artificial intelligence, and the optimization of audio output. We define some theories that our project is based on and outline additional research used to hit our weekly milestones as outlined in P0.

Foundational Theories and Technologies

- Ambisonic Sound: Ambisonics is a full-sphere surround sound format that goes beyond traditional stereo to capture the directional information of sound sources. This technology allows listeners to perceive sound in three dimensions, making it crucial for creating immersive soundscapes and acoustic holography.
- Acoustic Holography: This concept refers to the use of sound wave interference patterns to create a three-dimensional sound field. High-fidelity sound holograms can be constructed, allowing for an enhanced virtual reality experience where the listener is enveloped by sound seemingly emanating from specific locations in a simulated 3D space.

- Machine Learning in Sound Processing: Recent advancements in machine learning provide algorithms that can predict and generate complex sound fields, enabling the development of intelligent systems that adaptively shape audio to individual listeners and environments.
- Panning: Panning is the distribution of a sound signal into a new stereo or multi-channel sound field determined by the listener's position. It is essential in creating the illusion of movement and location for sound sources within an auditory scene.
- Frequency Filters: These filters shape the sound by allowing certain frequencies to pass through while attenuating others. High-pass filters let frequencies above a certain threshold pass, which can make a sound seem 'thinner' or 'farther away'. Low-pass filters do the opposite, allowing lower frequencies through and can make a sound seem 'fuller' or 'closer'.
- Bass and Treble: These terms refer to the lower and higher frequency ranges in audio. Bass enhancement boosts the lower frequencies, adding warmth and fullness to the sound, while treble enhancement brings clarity and definition to the higher frequencies.
- Reverb: Reverberation is an effect that adds the simulation of space to audio, creating echoes that make the sound seem like it's coming from a larger space, such as a concert hall or a cathedral. It's crucial for adding depth and atmosphere to the auditory experience.

Review of P0 Sources and Additional Research

1. Machine Learning for Sound Processing: While we didn’t successfully implement machine learning for sound processing for P1, we explored state-of-the-art techniques in deep learning, particularly those that can model and predict the behavior of sound waves in different settings to create an immersive auditory experience. We, however, aim to implement machine learning in our code (if needed) for the final submission.
2. Algorithms for Acoustic Modeling: By studying algorithms for real-time acoustic simulation, we gained insights into how to accurately replicate complex sound environments in real-time, which is integral to the final project as the goal is for the users to provide a link or a file that we can process and play with low latency.
3. Python Libraries for Audio Processing: Libraries like Librosa offer robust tools for audio analysis and are essential in building datasets required for training our machine learning models.
4. Real-time Sound Synthesis: Tools such as Max/MSP facilitate real-time sound synthesis and manipulation, crucial for developing dynamic soundscapes that respond to user interactions.

5. Optimizing Audio Output: Optimization techniques, including genetic algorithms, provide methodologies to fine-tune audio output configurations, ensuring the best possible listening experience.

There are additional links provided to research papers and documentation in the References page.

Methodology

Till P1, we were primarily centered around the development and application of sophisticated audio manipulation techniques using Python and several libraries, including Librosa, Sox, and SoundFile, among others. This section delves into the detailed methodology employed in the 8D Audio P1 code, articulating the steps taken to achieve our objective of transforming standard audio files into multidimensional auditory landscapes.

Installation and Setup: We began by installing the necessary Python libraries (sox. pydub, soundfile) to handle audio processing tasks. The following list summarizes the purpose of the libraries used:

- sox: A powerful command-line utility for applying effects to audio files.
- soundfile: A library for reading and writing sound files.

Audio Download and Preparation: Manually sourcing audio content, we obtained tracks from YouTube by direct download. These files were then transformed into a compatible format (WAV) for processing. This approach facilitated the accessibility of a variety of audio samples, which were crucial in evaluating the capabilities of our sound manipulation techniques.

Feature Extraction and Audio Manipulation: Through librosa, we analyzed the audio files to extract essential features such as tempo and beat frames, which played a vital role in guiding the audio manipulation process. The core of our methodology involved two innovative techniques to create an immersive audio experience:

1. Dynamic Audio Panning: We developed an algorithm to dynamically alter the audio's panning, creating a sensation of sound movement around the listener. This involved calculating amplitude modulation based on the song's tempo and applying it to create a rotating sound effect, simulating the "8D audio" experience. The sound was alternately emphasized between the left and right channels in a controlled manner, with periods of maintenance where the sound would remain steady before transitioning again.
2. Audio Effect Application: Utilizing sox, we applied several audio effects, including reverb, treble, and bass enhancements.

Simulation of Audio Elevation: An aspect of our audio manipulation involved simulating elevation changes through audio filters. We implemented high-pass and low-pass filters to mimic the sensation of sound moving vertically around the listener. This technique, based on altering the frequency response over time, contributed to the multidimensional aspect of the auditory experience.

Output Generation and Validation: The final step involved generating the processed audio output and validating its quality and format. We ensured that the output audio adhered to the expected parameters, such as sample rate and bit depth, and verified its compatibility with standard audio playback devices.

Implementation Details - 8D_Stereo.ipynb

We developed a Python script to convert audio files into "8D music," using the following process:

1. **Audio Downloading**: Utilizing yt-dlp, we downloaded audio files from YouTube, which were then converted to WAV format for further manipulation.
2. **Feature Extraction**: With librosa, we extracted essential audio features from the downloaded files, including the mono and stereo waveforms, as well as the tempo and beat frames.
3. **Dynamic Sound Rotation**: The core algorithm, implemented in Python, created a rotation effect of the sound between the left and right audio channels. This was achieved by:
    a. Amplitude modulation based on the song's tempo.
    b. Adjusting the volume of the left and right channels in opposition to create a sensation of circular movement.
4. **Audio Effects Application**: Utilizing sox and pydub, we applied audio effects such as reverb, bass, and treble enhancements to enrich the audio experience.
5. **Elevation Simulation**: High and low pass filters were employed to simulate elevation changes, creating a sense of audio movement along the vertical axis.

A link to our repo (curently work in progress): https://github.com/smriti06-lab/CS489
We do understand that calling it 8D is incorrect so the wording will be fixed for the final project. Upon further exploration of the github repository, you can see that we wrote significant amount of code (most of which isn’t shown) to see what works for our project.

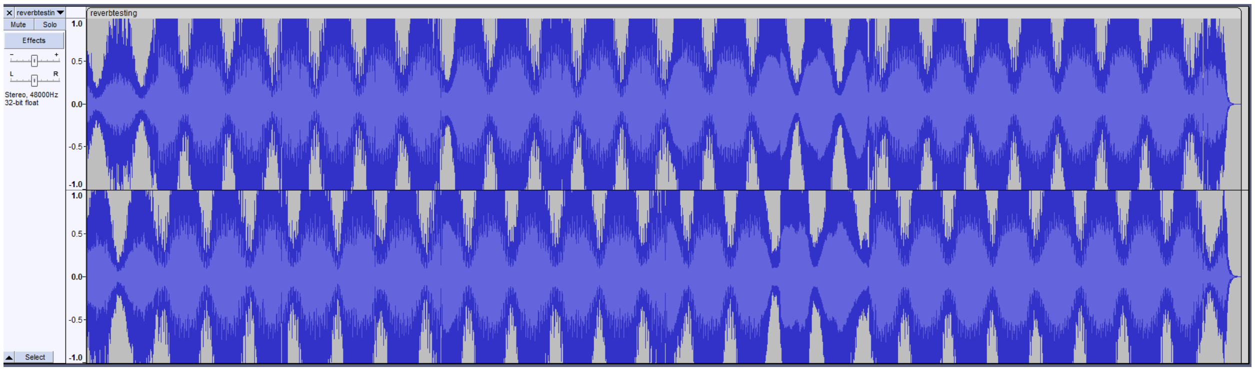
Results and Analysis

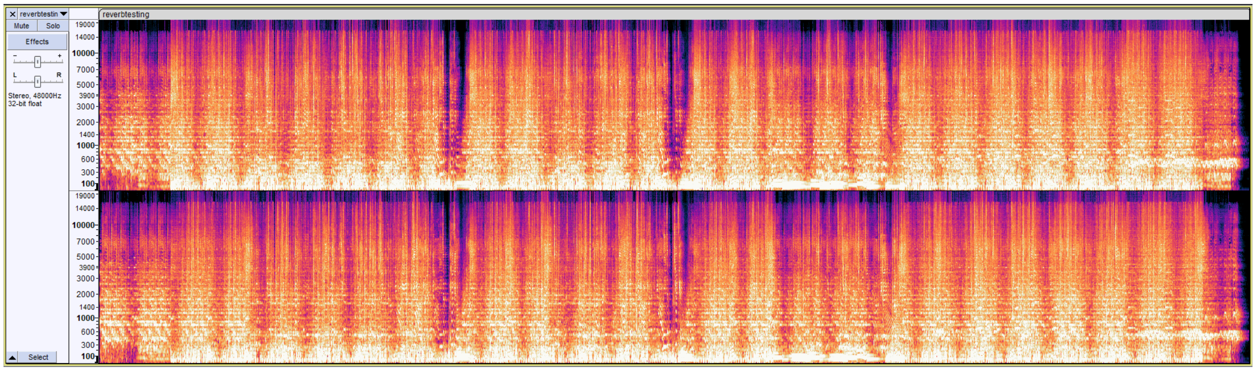
Our initial phase of testing focused on the manipulation of audio files through dynamic panning and the application of various audio effects. We employed standard earphones, including older models from the late 2000s, to evaluate the output and ensure broad hardware compatibility. The sample audio files used for testing were manually downloaded from YouTube, then processed through our Python script.

The conversion process involved shifting amplitudes alternately between the left and right channels of stereo audio. By doing so, we were able to simulate an "8D" auditory experience, creating the illusion that the sound was orbiting around the listener's head. The application of frequency filters further enhanced this effect, giving the audio a vertical dimension that contributed to the immersive quality.

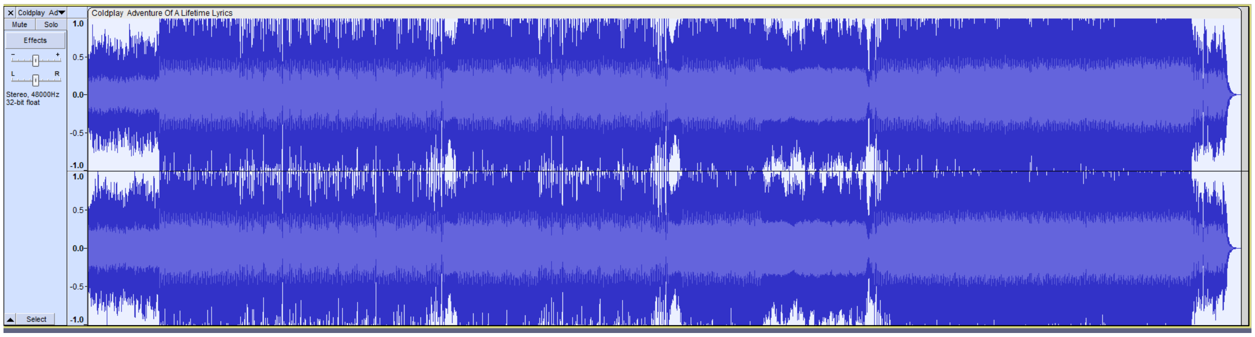
The addition of effects such as bass boost, treble enhancement, and reverb proved to be crucial in enriching the audio. Bass boost added depth and warmth, treble clarified high-frequency details, and reverb imparted a sense of space, making the sound appear as if coming from different environments, from intimate rooms to vast halls. These initial results are promising, indicating that our system can create a rich, immersive sound field with standard audio playback devices.

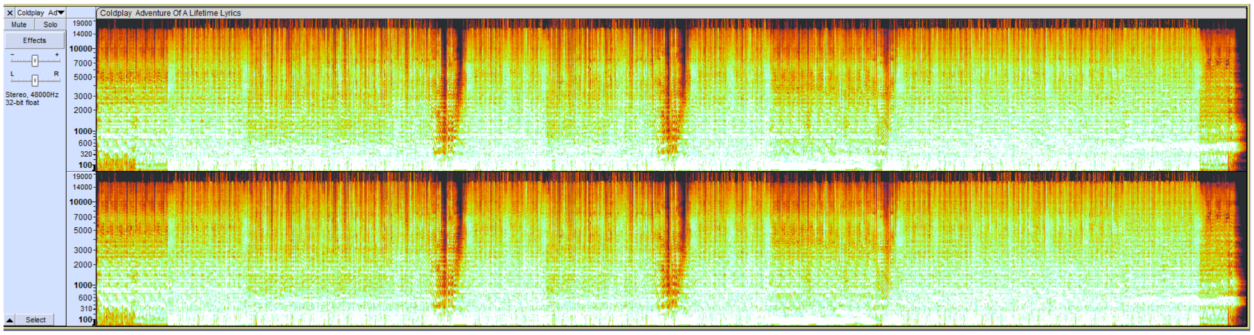
The provided images showcase the transformation of an audio file, specifically "Adventure of a Lifetime" by Coldplay, through our sound manipulation process aimed at creating an immersive 8D audio experience. In the first image, we observe the stereo audio file's waveform post-conversion. The alternating peaks in the left and right channels are prominently visible, illustrating the dynamic panning effect that has been applied. The regularity and symmetry of these peaks suggest a consistent and controlled transition of sound between channels, echoing the sensation of auditory movement that circles around the listener.





The second image presents the original waveform of the same song prior to the conversion. Compared to the post-conversion waveform, this one appears more uniform across both channels, lacking the distinct alternating pattern. The pre-conversion waveform indicates a typical stereo sound without the spatial manipulation that characterizes 8D audio. Below each waveform, a spectral frequency display provides further insight. In the post-conversion spectrum, there is a noticeable variance in intensity between the channels over time, aligning with the panning effects applied during the conversion. In contrast, the original spectrum shows a more even distribution of frequencies across both channels.





Future Work

As our project progresses, the next steps in the development cycle will address current limitations and expand upon our foundational work. We acknowledge that while we have explored machine learning models, their integration into our system has not been realized due to technical challenges, including file format inconsistencies and library incompatibilities. Overcoming these hurdles will be pivotal for our future work. Some of our major milestones for the following weeks include:

1. **Integration of Machine Learning Models:** The forthcoming phase will prioritize the integration of machine learning models. We aim to employ convolutional neural networks (CNNs) and recurrent neural networks (RNNs) to model sound wave behaviors and predict optimal audio field manipulations. This requires resolving existing issues with audio file formats and ensuring library compatibility, especially when dealing with backward compatibility for audio processing libraries.

2. **Testing and Fine-tuning Audio Processing:** A comprehensive testing and fine-tuning process is essential to refine the left and right sound movement in our audio manipulation algorithm. This will involve iterative adjustments to the amplitude modulation technique, ensuring smooth transitions and a more natural "8D" audio experience. User feedback will be integral to this process, informing adjustments that enhance the listening experience.
3. **Development of a User Interface:** We will design and develop a user-friendly web application to make our technology accessible. The application will allow users to upload audio files or provide links to online content, which our system will then process and convert into "8D" audio. This web app will provide a platform for easy interaction with our system and widen our user base.

Additional Areas for Research and Development

Looking beyond the current scope, we will explore the following areas for further research and development:

- Acoustic Environment Modeling: Delving deeper into acoustic simulations to recreate various environmental soundscapes that can be applied to virtual reality and augmented reality experiences.
- Latency Reduction: Improving real-time processing capabilities to minimize latency, which is important for synchronous audio playback. Given that audio files can be quite large, often measured in megabytes, the time taken for conversion can impact user experience. Therefore, optimizing our processing pipeline to handle large files more swiftly will be a focus area.
- Spatial Audio Algorithms: Advancing our algorithms to more accurately simulate sound localization and movement, enhancing the perception of depth and space.

Conclusion

Reflecting on the learning outcomes, the hands-on experience with audio processing and the challenges faced have enriched our understanding, pushing us to think creatively and critically. We applied theories from computational audio in tangible ways, seeing firsthand how adjustments in amplitude modulation or the application of frequency filters can alter a listener's experience. As we look ahead, we aim to integrate machine learning, enhance processing efficiency, and extend its accessibility through a user-friendly interface. Our journey thus far has been rewarding, and we are eager to continue this trajectory.

References

https://pypi.org/project/yt-dlp/
yt-dlp is a youtube-dl fork based on the now inactive youtube-dlc. It includes new features and patches while also keeping up to date with the original project.

https://pypi.org/project/soundfile/
The `soundfile` module can read and write sound files.

https://pysox.readthedocs.io/en/latest/index.html
pysox is a Python wrapper around the amazing SoX command line tool.

https://asa.scitation.org/doi/book/10.1121/1.406164
"3D Audio Technology" from the Acoustical Society of America.

https://arxiv.org/abs/1905.00078
“Machine Learning for Sound Processing" for an extensive view of deep learning techniques applied to audio.

https://research.nvidia.com/publication/2017-07_Real-time-3D-Acoustic
Nvidia's publication on "Algorithms for Acoustic Modeling" which explores the computational aspects of sound simulation.

https://librosa.org/doc/main/index.html
Librosa's documentation, a key Python library for audio processing.

https://www.tensorflow.org/tutorials/audio/simple_audio
TensorFlow tutorials for training deep learning models in audio processing.

https://docs.cycling74.com/max8/
Documentation on Max/MSP for real-time audio synthesis.

https://ieeexplore.ieee.org/document/8613755
Research on optimizing audio output configurations using genetic algorithms.

https://github.com/Zuzu-Typ/PyOpenAL
PyOpenAL for 3D audio processing in Python.

jiaaro/pydub: Manipulate audio with a simple and easy high level interface (github.com)
Pydub is used in the project to perform I/O operations on the sound files.