Anil Timbil

Ozgur Izmirli

COM312 – Digital Sound Processing

September 2, 2019

**Digital Sound Processing: Final Project**

As music industry keeps to improve continuously, 3D sound has a great potential in discovering new ways of hearing music. In this project, the problem was to create and analyze new expressions of an a capella song in a 3D space with different spatial configurations. We aim to see how individual musical sounds can be spread over space in order to create a better collective sound.

Due to availability of various individual voice parts, this project uses *Tupelo Honey* song from *Stairwell* album by *Co Co Beaux*. In each file there is only one person singing, and each person has a label for their voice part which can be identify as bass, baritone, tenor II or tenor I. The original song can be found on *Co Co Beaux*’s Spotify account for reference.

The framework of this project is based on our ability to place a musical monophonic sound to another location in 3D space. In order to accomplish the relocation of monophonic sounds, we used sound samples that were recorded during a 1998 study published by CIPIC Interface Laboratory at University of California. This study provides a set of Head Related Impulse Response (HRIR) data measured from a large group of subjects. Interpolation will not be considered in this project.

The HRIR data in this study has 25 azimuths, 50 elevations, and 200 samples over time. Azimuth values range from -80 to +80 degrees where the negative sign represents being closer to left ear and the positive sign represents being closer to right ear. More specifically, azimuth values follow these values: [-80 -65 -55 -45:5:45 55 65 80]. Elevation values, on the other hand, have more sampling ranging from -45 to +230.625 in steps of 5.625 where 0 represents directly ahead and 180 represents directly behind. One of the limitations of this data is the total number of azimuth and elevations. Although this data provides a list of possible combinations, we cannot be very precise about placing them to a specific location. Another limitation of this data is that it does not take into account the distance of the sound source. Although not addressed in this project, this problem can be solved adjusting the volume of the sound.

Another concept that is important to understand for experimenting with this program is the mode of spatialization. Modes of spatializations represent pre-determined azimuth and elevation combinations that represent real life scenarios. Although we could have allowed users to enter individual azimuth and elevation values for each sound track, this would not provide an easy comparison between different configurations. It would also not be user-friendly as users would have to make multiple calculations in order to place a sound in 3D space. Instead, this program relocates individual sound tracks based on pre-determined and easy-to-use modes of spatialization. A complete explanation of modes of spatialization can be found in the documentation folder.

In addition to modes of spatialization, there are three types of programs which can be selected in main.m file. The first program is called “original” as it mixes the input files into a mono sound without any spatialization. The second prorgam is called “example” which takes the input files and returns the example configurations based on different modes of spatialization. The third program is “spectral\_centroid” which is depended on the musical content as well as the mode of spatialization. This program calculates a spectral centroid value for each sound track, and sorts the signals in ascending order based on these values. With this approach you don’t need to pass in sound files in a certain order. Instead centroid calculation sorts them based on tracks’ brightness, and then follows the pattern of each mode.

In terms of technicality there are two important milestones to be explained. The modes of spatialization were coded as patterns consisting of the index number of azimuths and elevations. The file get\_pattern.m codes these modes of spatialization in patterns and assumes that they will always be in the order of bass, baritone, tenor II and tenor I. For the “spectral\_centroid” program this assumption is quite different in the sense that the program assumes patterns are ordered from less bright to more bright sounds.

Since the selected song has 13 sound tracks, modes of spatialization are designed to work best with 13 different sound sources. The “example” program, is specifically designed to work with this song to serve as a comparison mechanism for “spectral\_centroid.” If the number of sound tracks is less or more than 13, the program still works with a different pattern logic. In the case we have less than 13 sound tracks, the program will start placing the sound sources following the base-to-tenor order of patterns. Not all the spots in the mode will be filled with actual sound sources. In the case that we have more than 13 sound sources, the pattern will repeat itself in order to open up new spots for extra sound sources. Although this allows to have all sound tracks to be placed in 3D space, it may result in unbalanced configurations. It also only works in relative measures. If we have only sound sources of the same voice group, then this design may split this group into different voice parts although all of them might be originally from the same part.

Another milestone is about how we actually get the spatial sound from a set of inputs which includes azimuth, elevation, and sound signal. After passing these inputs, HRIR data for each ear were used to create the sound signal for each signal. Spatialization of a monophonic sound eventually created two channels, one for left ear and one for right ear. Repeating this process for each monophonic sound signal and adding each channel up outputs a final mix in which each original sound signal appears to be coming from another direction.

Applying the method elaborated above resulted in many sound files with different 3D sound characteristics. Each program has their own group of sound outputs. Original mix was included in order to provide a reference point for other programs. It was clear that 3D spatialization provided a very different sound than the original monophonic sound which comes from a single direction. We were successfully able to create different musical experiences of the same song placing sound sources in different locations in 3D space.

If you look at any of the sound files, there is a prominent blend between 3.33 and 3.41, and listening to this part we can more easily hear the differences between each mode. For example, original sound output gives a more pointed and concentrated sound. It is very hard to hear individual voices, and they clash in an unpleasant way. If we listen to arc\_1 of both programs, we can recognize the individual sounds are coming from different directions imitating a real a capella performance. Sound sources in arc\_1 are easier to distinguish, and they blend better as there are more spaces among them.

Due to changing interests and tastes, it is difficult to decide if a given output file sounds better than others. Depending on musicians’ preferences and groups’ needs, these different outputs can be handy and useful. Therefore, it is up to users to decide which mode of spatialization sounds better.

Another interesting result was to see that the “spectral\_centroid” program was able to provide very similar results to our example configurations. Calculating spectral centroid values provided a good method for distinguishing voice parts (bass, baritone, tenor II and tenor I), and allowed me to sequence the sound sources relative to each other. Thus, it was possible to get the bass-to-tenor order which is required for modes of spatialization.

In conclusion, this project allowed me to experience with different configurations of 3D music in a creative way. Even with a limited number of output files, we were able to hear different expressions of a song which could be suitable for different occasions and purposes. A variety of groups, not limited to acapella, can benefit from this implementation to experiment with their choreography in advance, and they can save a significant amount of time during practices. With further improvements, such as adjustable coordinates within a song, we can explore 3D sound further to create unique musical experiences.