Final Year Project Report

# Synthesising the Sound of Crowds

Eoghan McDermott

A thesis submitted in part fulfilment of the degree of

**BSc. (Hons.) in Computer Science**

**Supervisor:** Dr Fred Cummins

UCD School of Computer Science

University College Dublin

April 2019

# Project Specification

## General Information:

Crowds emit sounds that are informative about their collective activity. When voices synchronise in chant, it quickly becomes clear whether we are dealing with a pious assembly in prayer, an enraged mob demanding change, or an enthused bunch of football supporters. This is evident from musical properties of the collective voices alone, without reference to the specific words employed.

In this project, you will use multiple audio recordings of individual voices, and combine them in ways that are suggestive of a crowd of protesters, in one case, and of a liturgical celebration, in the other. So an initial requirement is that you be capable of carrying out some fairly simple mixing of audio files, that is, of taking multiple sources and combining them.

You will develop means to explore the contribution of rhythm, timing, loudness, melody, and randomness to parametrically vary the overall effect, from protest to prayer. This will require listening to crowd sounds, and consideration of the acoustic properties that might signal purposes in one way or another.

Many examples of collective speech are available at jointspeech.ucd.ie. This project will be best suited to a student with some experience of audio mixing in a musical context. For sound mixing, Matlab or Java are options, depending on the experience of the student.

### Core:

You will record (or collect) multiple source recordings of voices. The actual words being spoken or chanted are not important, and we will be as happy using "rhubarb, rhubarb" as any other text. For individual recordings of single voices, you will do some preliminary analysis to mark significant onsets in time, to describe the pitch and intensity variation. You will consider the use of the Praat programme (www.praat.org) to produce modified recordings with stylised pitch contours.

You will construct a system that allows you to overlay and mix these voices in various ways. Elements to be included will be the timing of offsets, the degree of pitch stylisation, and the general level of randomness.

### Advanced:

Given a set of input voices, synthesize them to suggest (a) a crowd of protestors, (b) a crowd of worshippers, and provide parametric means for warping the sound from one form to another.

# Abstract

A great deal of work has been done on the synthesis of individual speech, as well as on the idea of joint speech, which deals with collective voice. However, past study of joint speech has dealt with very rhythmic speech, chanting and singing etc. The idea behind this project, on the other hand, is concerned with the synthesis of a somewhat more unintelligible crowd, which is not something that has been explored before. This report documents the developing of a system that, through the combination and overlay of individual audio recordings seeks to explore the landscape of ambient, unsynchronised crowds. Through the means of parameterisation, this system will be able to synthesise varied, and unique sounding crowds, and can be used as a means to explore the place of these crowds within the umbrella of collective speech.

# Table of Contents

# 1 Introduction

This project is focused on the sounds of crowds, and explores their different characteristics and structures, which vary depending on the particular crowd in question. Through research conducted over the course of this project, the goal is to identify these characteristics and use that information to gain insights about the overall sound of a crowd.

As these ideas are explored and investigated, a system is to be built which can synthesise the sound of a crowd in such a way that the various crowd traits can be parameterised and varied to create crowds that are unique sounding and perhaps even fit a specific purpose.

It is important to note that this project diverged from its original goals. Initially, the project aimed to focus on two specific types of crowds, on identifying and synthesising the sound of a crowd of protestors and a crowd of worshippers. These two different types of crowds were to be examined in depth as they lie at different ends of a sort of spectrum of crowd sounds and would hopefully possess some unique and interesting characteristics. As well as this, they are two cases that are very socially relevant in the midst of a society rife with civil unrest and clashing ideologies. While these crowds are important, it is worth noting that they are only two instances within the substantial domain of collective effervescence. (Durkheim, 2008/1912)

Early into the lifecycle of the project, however, the joint decision was made between supervisor and student, not to limit this project's scope to only these two specific crowds. Instead the project would focus on crowds of an ambient nature, a form of unsynchronised speech, exploring the sounds of unsynchronised crowds. Through study and research the project aims to gain insight about the traits and overall gestalt of these ambient crowds. This can then be used as a basis for a set of parameters, which can be varied to recreate the sounds of various different unsynchronised crowds.

This project is an of an investigative nature. The end goal of the project is not to produce a commercial product that generates a variety of different crowds. The goal is to explore the domain of unsynchronised crowds and that of collective voice.

A major component of this project is concerned with the finding a set of parameters which can be varied to synthesise the sound of different crowds. It is important to note that as this is an under researched area. Upon beginning this project, these parameters were unknown and finding out what form they could take lies at the crux of the work of this project. This project served as a vehicle for exploring a number of possibilities, and the parameters implemented in the project served their purpose well as a means of investigating the domain of unsynchronised crowds. If in future this project was extended to fit a specific use-case, other parameters could prove to be of more use than the ones implemented here.

This report outlines the process of creating such a system for synthesising the sound of crowds. It details the initial research undertaken to understand the project and where it lies in the larger domain of collective speech, as well as the exploration of the parameter space to vary the produced crowds, before delving into the design and actual implementation of the system, and how it developed and was evaluated.

# 2 Background Research

As preparation for this project, a great deal of time was spent listening to and studying crowds and discussion was had between supervisor and student, trying to identify different characteristics of a crowd and how that variation could be recreated and implemented in the system being developed. In order to be able to synthesise a crowd with any degree of authenticity, one must first become familiar with the sound of crowds. A wide spectrum of different crowd sounds were examined, covering such extremes as a group of murmuring worshippers all the way to violent and angry protest marches, and a variety of others in between. Thankfully, today's media provides an ever expanding wealth of resources that can be used to acquaint oneself with the sound of crowds, and to build up the knowledge base necessary for the work entailed in this project. Without spending time immersing oneself in crowd sounds and the wider domain of collective speech, a project of this nature would be impossible to implement.

A key element of the aforementioned domain of collective speech is joint speech.

*"Joint speech is speech produced by two or more people who utter the same thing at the same time."* (Cummins, 2019)

Joint speech deals with the synchronised speech of a group of people. Through its study a number of insights have been gained into the nature of crowds and how they function. However, while these insights are valuable, the study of joint speech  falls short in that it is only concerned with synchronised speech.

A prime example of joint speech that aligns closely with the scope of the project is Hans Zimmer's soundtrack for the film *The Dark Knight Rises* (Nolan, Thomas, & Roven, 2012). In order to create the sound of a chanting crowd, Zimmer reached out to fans and invited them to record themselves chanting and to contribute their voice to the crowd that he was creating. The crowd he produced are all chanting in unison, and embodies what joint speech sounds like. Not only is this an example of joint speech, but the process with which he created this crowd aligns directly with the aims of this project, a crowd composed of individual voices, all recorded separately and then layered together to form something greater than the sum of its parts.

Synchronised speech and crowds are very closely linked, but to think that the speech in the sound of a crowd must be synchronised would leave a vast swath of crowds out of one's scope – those crowds that have unsynchronised speech.

This project seeks to examine and to help to fill that gap in knowledge on unsynchronised speech in crowds. While the area is largely unexplored, some research has been done in the domain of unsynchronised speech. This comes in the form of babble.

Babble is the name given to the general hubbub of background speech. With sound like this multiple voices can be heard, however it is very difficult to actually discern any individual voice or words from the overall din of the crowd in question. The words spoken by each individual in the crowd are unimportant, in fact, typically a simple repetition of "rhubarb, rhubarb, rhubarb" is enough to contribute to the sound of a crowd such as this. Together, however, the group of voices form more of the sum of their parts, and the incomprehensible babble sound is created.

*"It has been said that the best place to hide a leaf is in the forest, and presumably the best place to hide a voice is among other voices"* (Miller, 1947, p. 118)

Babble is ubiquitous in everyday life and there exist countless examples of it in media. Take any average film or television show that contains a scene out on the street or in an office environment say, and the hum of babble will be present in the background as the main characters speak and interact. Babble, by nature, is a form of unsynchronised crowd speak. By definition it is not overly interesting or unique. While work has been done on the subject, research about babble is mainly concerned with the difficulty in discerning individual voices from a group of people speaking at once, known as the "cocktail party effect." Unfortunately, one does not gain many useful insights into the nature of crowd sounds themselves through babble. As such, this project is delving into relatively unexplored territory as it seeks to examine the sound of a crowd in general, rather than the voice of an individual in a crowd.

The sound of crowds can be seen being used for academic and scientific purposes as background noise and an alternative to the likes of white noise in the study and research of hearing. Psychoacoustics is the branch of psychology concerned with the perception of sound and its physiological effects, how humans perceive different sounds. A psychologist was one of the possible use-cases considered during the development of this project. The system created could be used as a means of creating crowd sounds for use in psychoacoustic experiments.

The bustle and hum of crowd sounds is also used for more artisanal purposes such as in film and television, or at concerts and other events. Another use-case kept in mind during the project was that of a film director. The system could be used to synthesise various crowds to serve as a background for real actors, giving more control to the director than dealing with a large group of extras in person, and also possibly to reduce costs.

The crowds seen in the aforementioned media and the activities that would of most relevance to this project would probably be crowds whose purpose is to provide a backdrop of a general background hum and ambiance for the main characters. However, crowds of a more direct nature, with an active role and purpose are widely seen too. These such crowds would tend to fall more into the domain of the work done on joint speech, as they tend to make use of synchronised speech. That is not to say, however, that they are entirely irrelevant to the work done in this project. This project and the work done in joint speech both fall under the domain of joint speech, so it is not impossible for an insight gained in one to apply to the other.

The system created in this project produces crowds composed of a number of individual voice samples all woven together.

In technical terms, the samples used have a sampling rate of 44.1kHz and are stereo, which is a de facto industry standard when it comes to dealing with audio. It is important that all of the samples in the sample library have a uniform format, as any inconsistencies could cause issues and errors when trying to manipulate the files while creating the crowd sound.

This de facto standard of 44.1 kHz arises from the fact that the human ear is capable of hearing sounds in the range of roughly 20 Hz to 20,000 Hz. (Rosen & Howell, 2011)

When this, along with the Nyquist-Shannon sampling theorem, which says that the sampling frequency must be twice as great as the maximum frequency to be produced, a sampling rate of more than 40kHz is needed to properly capture and represent sound. (Shannon, 1949)

While this information may seem unimportant and overly technical when examining the sound of a crowd on a macro level, it is important to understand the foundations on a micro level  too, so that there are no gaps in understanding when  examining and manipulating the individual samples that the crowd in question is composed of.

If there are any discrepancies in formatting between the different samples used, the system that has been developed will throw an exception and stop running, so it is very important that all of the samples adhere to the proper format.

In order to create the variation necessary to synthesise unique-sounding crowds, one could manipulate the individual voice samples used to form the crowd. Apart from directly manipulating the samples through the Java Sound API, there exists a program called Praat, which was designed with low level audio programming in mind. With Praat, one can manipulate an audio file in a variety of ways, including modifying the gain and intensity, but also through the use of filters. During the early stages of the project, such signal processing was experimented with and applied to a number of files. It was found, however, that this manipulation of audio introduced undesired artefacts, which reduced the quality of the crowds being produced.

After a great deal of consideration, and with this artefact problem in mind, the decision was made to gather a large library of samples and organise them rather than manipulate a smaller set of samples. In this project time was spent organising the files into different categories and folders, e.g. soft, loud, etc. By doing this the system can draw more from a given folder to fit its parameters. For example, one of the parameters that was implemented was a soft to loud dynamic. Depending on what the user wants, the system can draw more heavily from a collection of soft samples or a collection of loud samples, as necessary. The other option would be to manipulate a smaller set of samples to fit the given parameters. As mentioned above, audio files can be manipulated in a number of ways, for example, pitch and gain manipulation. To use again the example of the soft to loud parameter, an individual file could be altered to sound softer or harsher as needed.

The decision to opt for organising the samples beforehand rather than manipulating them in real-time was made due to its proven effectiveness, which has been seen in the field of concatenative speech synthesis. In a somewhat similar manner to this project, concatenative speech synthesis combines individual audio clips together to form a new sound. In concatenative speech synthesis individual syllables are chained together to form words and sentences, as opposed to individual voices making up the sound of a crowd in this project. It was found that whenever an audio file is manipulated, artefacts are introduced. These artefacts, while often appearing negligible in terms of a single audio file, reduce the quality of the sound produced when many files are added and chained together. By instead drawing from a larger set of individual clips, as was done in this project, better results are produced.
Apart from being more computationally efficient, the output is more reliable and devoid of any aforementioned artefacts.

# 3 Parameters for Crowd Synthesis

It has already been established in this report what specific types of crowds are to be created with the system developed for this project, ambient, unsynchronised crowds; made up of individual voices interlaced together. This domain encompasses a wide range of crowds, which begs the question – how could one possibly find a set of parameters that would be able to recreate the variation seen in this space?

The process of finding such a collection of parameters is much more difficult than one would initially expect. As this is a relatively unexplored research area, there is a lack of prior work to serve as a guide when trying to decide on the parameters to use. A number of parameters were selected for implementation in this project, and proved very effective in recreating the desired variation necessary to synthesise the sound of the ambient, unsynchronised crowds. These parameters were discovered after spending a great deal of time listening to and discussing the sounds of numerous crowds, what made them unique, as well as some common threads that linked them all. While these particular parameters are very important in the scope of this project, other parameters might prove to be more useful, depending on the domain being explored.

The first parameter that was explored was a ratio of male/female speakers. By varying this ratio one can capture the sound of different groups. Within society, distinct groups have different make ups, and often times the ratio of men to women is the most explicit difference between these groups. Numerous examples exist of this distinction, for example, crowds at Premier League football matches in the UK are largely made up of men. Changing the value of this parameter allows for a variety of different and unique sounding crowds to be created. Perhaps the earlier mentioned use-case of a film director would want to synthesise a crowd of a right-wing nationalist group, such as those seen with the rise of the alt-right in the US, and the likes of Tommy Robinson in the UK. To create such a crowd, being able to skew it to consist of a majority of men would be invaluable in creating an authentic sounding crowd. To turn attention to the psychologist use-case, this parameter allows for experiments along the lines of identifying lone male voices in a crowd of female ones, tying in with other research done on babble and the "cocktail party effect," which was discussed earlier in this report.

Another parameter that was investigated was the actual number of voice samples in the crowd. After listening to numerous crowds a clear difference between them was the number of people in them, the size of the crowd. By being able to control the size of a crowd one can change the overall sound completely. Even if the other parameters used have the same values, a crowd of 10 people will sound completely different to one of 100 people. In the system the lower and upper bounds on the number of voice samples was found after a deal of experimentation. For the lower bound, the number of samples was varied until the sound produced changed from individuals speaking one after another in a synchronised fashion, to a group of people speaking over one another in an unsynchronised manner, the sound of a crowd. The establishment of a lower bound makes sense, but why bother finding an upper bound? While yes, in theory, a crowd can always grow larger, in practice, once a certain threshold of voices is reached, the crowds produced are purely white noise, with no voices or words distinguishable. This is not the type of sound this project seeks to recreate, and so an upper bound on the number of voices used is necessary. After a number of experiments with increasing amounts of voice samples used, an upper bound was found for this parameter. To again mention the use-case of a film director, by controlling the number of individual voices in a crowd, one can create crowds that are quite sparse and small or large and dense so as to fit a particular setting, for example, a roomful of people vs. a stadium full of people.

The final parameter that was used in this project was a dynamic of soft vs. loud or harsh voices. This parameter was chosen as it can be used as a means to capture the tone and emotion of a crowd. Using the earlier mentioned technique of organising the set of voice samples, the system can draw more from the appropriate library of samples when synthesising a crowd. The “soft” voices used are as one would imagine, softer than the average voice. This comes from having a collection of voices that are relatively soft-spoken, as well as whispering and quieter voices. These samples are effective in capturing a calm, relatively happy, tone. The “loud” voices are not just voices with higher than average volume. These voices are of a harsher nature, and include shouting, and are generally voices of a more aggressive nature, lending quite well to conveying anger and frustration. The ability to control the tone and emotion allows one to easily convey the overall sentiment of a crowd. This is of great value, especially for the use-case of a film director. By skewing a crowd to consist of mainly softer voices, one could synthesise something akin to a crowd of worshippers. Similarly, by opting to use only harsher, louder voices, a crowd of protestors could be generated.

Through the interplay of these different parameters, a wide spectrum of different sounding crowds can be created. These particular parameters and the dynamic between them allow for exactly the kind of crowds that were in mind upon the project’s inception to be synthesised, providing a means of producing unique and interesting output. As has already been mentioned, if further work were to be carried out to tailor this project to suit a particular use-case, as opposed to the general approach used here, other parameters could possibly prove more effective than those mentioned above, when trying to capture and recreate the specific characteristics and traits necessary for that particular use-case.

# 4 Design & Implementation

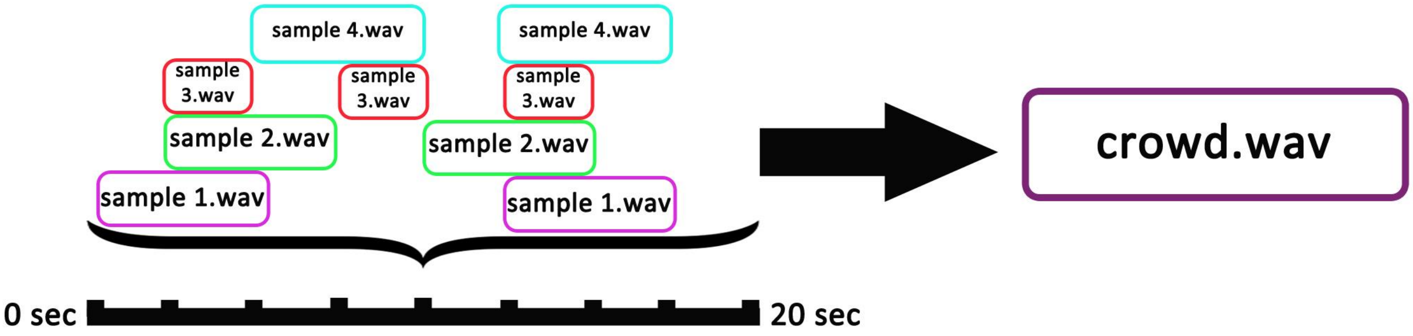
This project is focused on the sounds of crowds. As such, working with audio is a central tenet of the project. Before beginning work on any actual implementation of the crowd synthesis system, a suitable language needed to be selected, with which one could suitably and easily deal with audio. In order to work efficiently, the right level of abstraction needed to be found. One needs to be able to manipulate the audio data to a certain degree, but not get lost in minute technical details. A great variety of resources are available for audio programming at this relatively low level. Certain languages, for example, C, give the developer access to resources all the way down to the machine level. While this can prove very useful when manipulating audio and other data, in terms of the scope of this project, it doesn't quite fit the desired abstraction level. Other programming languages possess resourceful libraries that allow one to perform a number of complex operations without getting lost in the low level semantics of it all, while still providing the ability to deal with data at a byte level if desired. Two prime examples are the Java Sound API and the Python Multimedia Services Library. After some deliberation and investigation, the decision was made to write the project in Java and make use of the aforementioned API, as it appeared to be more versatile and robust than the other available options, and a good fit for the previously mentioned problem of finding the appropriate level of abstraction.

Obviously, for this project to work one needs access to the audio data from each individual file. By using the Java Sound API that process was made simple and efficient. A .WAV file is made up of a header component used to store assorted format information about the file, followed by the actual audio data itself. The size of these headers can vary from file to file, but thankfully, rather than having to be concerned about where exactly the audio data in file is stored, the API allows the user to work solely with the audio data itself (and also to easily access the formatting information, such as the number of channels, etc.)

The crowds that this system produces are made up of a combination of individual voices. In order to create these crowds a buffer is used, in which the individual voices are combined and overlaid. The individual audio from each file is added at a byte level. As mentioned earlier, the individual voice files are stored in 16-bit .WAV files. So every "unit" of audio that is being dealt with at a time is 16-bits long. Each of these individual values can be stored in Java's *short* primitive data type, which is conveniently also 16-bits long. In order to avoid the problem of byte overflow, which could quite feasibly occur when overlaying numerous values, the buffer itself uses the *integer* data type. 32-bits provides ample headroom to avoid the overflow problem. However, in order to make sure that the synthesised crowd can be saved in a new .WAV file, which is also 16-bit, the values stored in the buffer might need to be scaled to fit within the bounds of the aforementioned

.WAV container. This scale factor can be found by finding the highest value in the buffer and calculating the difference between it and the largest value a 16-bit *short* can hold. This number can then be used to scale the other values stored in the buffer. Similar to how the Java Sound API allowed efficient access to the audio data of a given file, it also makes the process of creating a new audio file a relatively smooth one. It allows one to format the file as desired and then pass it the actual audio data, saving time and avoiding having to create the header component of a .WAV file by hand.

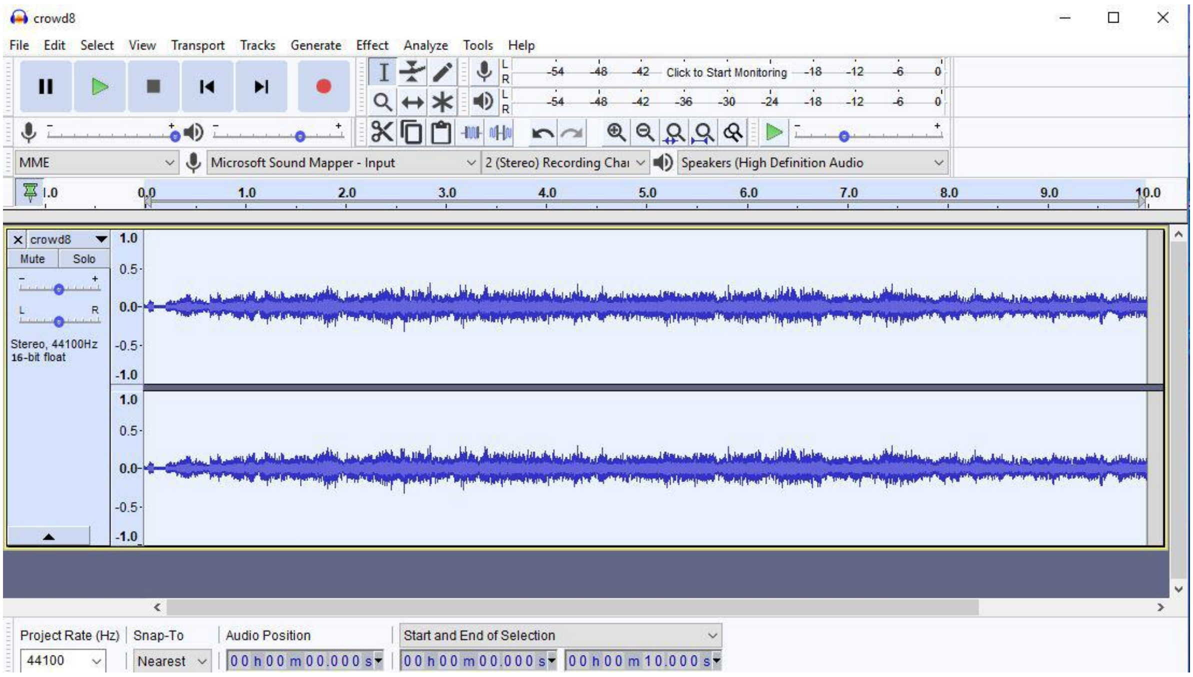
Fig. 1 - Buffer Visualisation:



The individual files are randomly overlaid in the buffer, which is done using a method called *randomiseOffset()* . The decision was made to uniformly distribute them throughout the buffer, in order to produce a natural sounding crowd. A problem that arose during development was that when two voice samples were laid over each other, the crowd produced was inaudible and full of grating static noise. This problem stemmed from a disparity between the conceptual level of working with audio samples and the lower level of how this was actually implemented, the data types that were being used to store said samples. The mapping of audio samples to actual bytes of data in the buffer needed to be adjusted so that the two levels of abstraction were in line with one another, and the data was stored in such a way that it properly mirrored what was happening at a conceptual level.

In order to create organic and natural sounding crowds, the system needs to be able to draw individual voices from a library that is rich and varied. There exist as many crowds as there do people, so a wide range of individual voices are needed to produce worthwhile output. The individual voices used in this project were taken from a variety of sources, including TV and radio, online content, as well as the more traditional recordings of individual people. The program Audacity allowed for individual clips of audio to be gathered from a larger source file, for example, extracting snippets from a speech.

Fig. 2 - Audacity



crowd8

Stereo, 44100Hz
16-bit float

Project Rate (Hz): 44100

Snap-To: Nearest

Audio Position: 00 h 00 m 00.000 s

Start and End of Selection: 00 h 00 m 00.000 s 00 h 00 m 10.000 s

The word variety is key to building a proper library of voice samples. While the individual words spoken by a particular voice are largely irrelevant, some other linguistic properties prove to be of much more value. value. Prosody is concerned with patterns of rhythm and sound in speech. It deals with things such as tempo, tone and intonation. By gathering samples on the basis of possessing unique and interesting values for these properties, rather than the specific words being spoken, a more varied and rhythmically richer sample library can be created, ultimately leading to more realistic and authentic crowd sounds.
As well as looking at prosody and its related attributes, it is important to capture a broad dynamic range of samples. The library needs to contain voices that are quiet and whispering, as well as those that are louder and shouting.

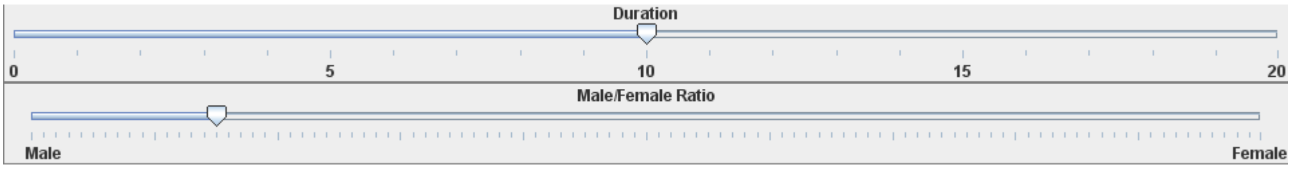
A crowd is made up of individual people. When these individuals are speaking, they are all located at different distances from the listener, and so will appear to be speaking at different volumes. In order to mimic this effect, samples with differing levels of volume needed to be gathered. The aforementioned program audacity provides an option to change the volume of a file, so this variety in volume was added in by hand. So within the library exists samples that, although they are similar in terms of dynamic range, they have different volumes. For example two people speaking in a similar tone but one is much louder than the other.

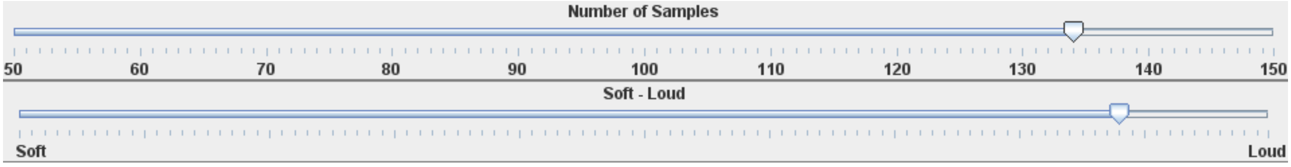
As has been discussed earlier, a key aspect of this project is the exploration of the parameter space that can be used and varied to synthesise the sound of a crowd. In practical terms, a GUI needed to be created to allow users to vary the parameters that were selected for implementation, so that they can create novel and unique sounding crowds. Java's Swing components aid this process, providing implementation of things such as buttons and sliders. The GUI created for this project wasn't focused so much on aesthetics so much as it was providing users a simple means of playing with and changing the parameters used to shape the sounds of the crowd that the system produces. The GUI features a number of sliders that allow for this variation, as well as showing the user what individual voices are used to make up the larger crowd and where these voices are drawn from. This information could be useful in identifying particular elements or characteristics that produce novel and interesting output. The GUI allows the user to playback the newly-created crowd and the generation of further crowds, each different to the last.

While not a real parameter itself, being able to vary the length of the actual crowd audio file produced, i.e. how long the crowd is speaking for, is an important and practical feature. The user should be able to decide on the duration time that the generated crowd is speaking so as to best suit their needs. To do this a slider was added to the GUI, similar to those used to vary the different parameters for crowd synthesis.

Seen below are the earlier mentioned parameter sliders which can be varied and changed to create the sounds of different unsynchronised crowds.

Fig. 3 & 4 - Parameter Sliders:





# 5 Testing & Evaluation

There exists no mathematical formula for evaluating the sound of a crowd. As such, there was no one simple test that could be used to gauge the overall quality of each crowd that was synthesised. Throughout the project a large amount of collaborative analysis took place. Individual crowds were listened to and discussed at different points in the project's lifecycle, highlighting qualities and elements that needed further work and noting what sounded authentic.

One choice that was influenced by this collaborative analysis was the earlier discussed problem of whether to manipulate a smaller set of samples or organise a larger sample let with no manipulation of the samples themselves. By creating crowds which had different types of signal processing applied to its sample base, and comparing them to crowds free of manipulation, the issue of artefacts causing a reduction in overall quality was clear to see, informing the decision on which was the superior route to take.

Initially, the system implemented was rudimentary and only capable of synthesising monotonous crowds without any great deal of variety. However, over the course of the project, the crowds that the system could create improved greatly. This improvement was twofold.

Firstly, through continued expansion of the sample library of individual voices from which the crowd was generated, better sounding crowds could be produced. More variety in the sample base led to unique and interesting output and in general, the crowds produced sounded less monotonous and more organic than those it was initially capable of creating. This variety stemmed from various discussions on what kind of samples were interesting or worth adding to the sample library, which influenced how further samples were collected, and what was to be avoided.

Secondly, as more parameters were added, the variety of crowds capable of being produced increased exponentially. The decision on what parameters were worth implementing was only reached after a great deal of collaborative analysis. After listening to a variety of crowds, it was jointly determined that collective emotion and crowd demographics were very important. This led to the selection of the two main parameters implemented, the male/female ratio and the emotive soft-loud dynamic.

As has already been mentioned, the parameters are an essential aspect of the project. For example, with the parameters which were implemented in this project, the system is capable of creating everything from a small group of murmuring women, to a massive crowd of loud and boisterous men, and everything in between.

This project is an exploration of what makes up the sound of a crowd. As the robustness of the system improved, so too did the amount of analysis done on its output. The addition of the parameters allowed for much more experimentation to be done and to better identify what elements were adding to or reducing the overall quality of the synthesised crowds.

# 6 Conclusions & Future Work

This project has achieved what it initially set out to do, construct a system that allows the sound of a crowd to be produced from individual voices, with parameters that allow the overall sound of the crowd to be warped. During the process, however, a whole new avenue of potential research has been opened up – the parameterisation of a crowd, what makes a crowd sound like a crowd?
That is not to say that the project is incomplete, but rather that it has laid a foundation for further work to be carried out in this space.

A crucial element of this project was the exploration of the parameter space. It is important to note that the parameters that were implemented in this project are not necessarily the only possible ones. There is no set list from which one can draw these parameters from. The parameters used in this project were selected after spending a great deal of time listening to various crowds and trying to grasp their overall gestalt and characteristics. If this project were to be modified or extended to fit a single use-case, rather than the more general approach taken here, other parameters than those used could prove more worthwhile to the user.

As such, there is always more work that can be done in further exploring this parameter space. Some possible parameters include, but are not limited to, the density of how voices are distributed in the crowd, a more rhythmic distribution could produce something very different to the current uniformly distributed approach, as well as some measure of the abruptness of individual voices used in the crowd. Different use-cases could benefit more from some of these parameters than others.

Apart from expanding work on the parameterisation of a crowd, more work can always be done on the sample library that the system draws from. While the system currently has quite a broad range of samples, this can continually be updated and added to, with each addition contributing to more authentic and organic sounding crowds being created. The further exploration of the parameter space, along with a continually expanded sample library are general improvements that can be made.

To better suit the earlier mentioned use-case of a psychologist, more precise controls might need to be added, such as sound pressure level. This, along with other, more psychoacoustic-orientated controls could allow for more varied experimentation through a psychological lens.

If this project was altered to be of more benefit to the other use-case mentioned earlier, that being a film/media director, the precise control of specific audio elements might not be necessary but adding some other general options could be of much more use. An aspect of crowds that this project does not cover is their location. A crowd in a busy market might possess a different tone and other qualities when compared to say a crowd at a football match or a protest march. As well as this, while the words spoken by an individual spoken in a crowd aren't of great importance, it might be of use to a director to be able to compose crowds where everyone is speaking a specific language. The individual voices in this project are for the most part speaking English, as more emphasis was put on their acoustic properties rather than the language being spoken. For a film director, having the option to select the spoken language could prove very useful.

This project took a general approach in exploring the parameters necessary in order to recreate and synthesise the sound of various different unsynchronised crowds.  While there are other possible domain-specific parameters that could be implemented to fit a specific use-case, the parameters used here are what is necessary for the scope of this project. Any further parameters added would depend on the specific end-use application in mind.

# 7 References

Cummins, F. (2019). *The Ground from which We Speak: Joint Speech and the Collective Subject.* Cambridge Scholars Publishing.

Durkheim, E. (2008/1912). *The Elementary Forms of the Religious Life.* Courier Corporation.

Miller, G. (1947). *The Masking of Speech.* Psych. Bull.

Nolan, C., Thomas, E., Roven, C. (Producers), & Nolan, C. (Director). (2012). *The Dark Knight Rises* [Motion Picture].

Rosen, S., & Howell, P. (2011). *Signals and Systems for Speech and Hearing* (2nd ed.). Leiden: BRILL.

Shannon, C. (1949). Communication in the Presence of Noise. *Proceedings of the IRE (Institute of Radio Engineers), 37*(1), 10-21.