UNIVERSITY OF DERBY
DEPARTMENT OF ELECTRONICS, COMPUTING &
MATHEMATICS

A PROJECT COMPLETED AS PART OF THE REQUIREMENTS
FOR
BSC (HONS) COMPUTER SCIENCE

Augmented Creativity:
Can chord recognition software be accurate enough for a complementary computationally generated tune to be created?

Sophie Berry
April 28, 2017

ABSTRACT

This paper reviews the possibility of creativity being augmented through the use of a chord detection algorithm. In particular, it attempts to prove that the accuracy of a chord detection algorithm can be high enough to allow for the creation of a complementary melody. Software is written to test this theory, which uses concepts currently found throughout music composition and within audio related software, such as Fourier Transforms and ADSR Envelopes. The analysis of multiple questionnaires allowed for the proof that chord detection algorithms can be accurate enough to be used in melody creation.

TABLE OF CONTENTS

1. INTRODUCTION

1.1. Project Rationale

Augmented Creativity refers to the concept of improving the levels of creativity that are currently available. In terms of software, this could be a computer creating something which previously would have been done by a person, possibly because the intelligence of a computer was not enough to complete creative tasks.

Recently there have been advances into the world of augmented creativity with computers, particularly in terms of music. Some of these revolve around creating music entirely through the use of a computer, such as the generation of entirely new songs that use stylistic features from existing songs (Sony CSL, n.d.). While some of these take an input and add to this existing input to improve upon it. This includes software which creates a backing track to accompany a tune played in, such as in RockSmith's Session Mode (Rocksmith 2014 - The Fastest Way to Learn Guitar, 2014), or as in MySong, which takes a vocal melody input (Microsoft Corporation, 2009).

However, the existing software makes use of specialist hardware, such as RockSmith's use of a RealTone cable (Ubisoft Entertainment, 2013), or algorithms that look at the pitch of a melody (MySong). There is currently no software available which is able to use chord detection algorithms in order to create music. This greatly limits the ability of music processing software, as it requires single melodies to be played in, rather than chords – something that would be much more beneficial when determining the creative abilities of artificial intelligence.

For this reason, this project will look at the concept of using chord detection algorithms to allow for the creation of computationally generated melodies. There have been a small number of research papers that discuss the process of chord detection algorithms (mentioned in detail in Section 2.2), but thus far, they have not applied these algorithms and determined their effectiveness in conjunction with melody creation. Therefore, the next logical step in this research would be to prove that using the characteristics found via chord detection a melody could be successfully added to the input.

1.2. Project Aim and Objectives

The overall aim of this project will be to discover if a chord detection algorithm can be accurate enough for the creation of a complementary computationally generated melody.

In order to achieve this aim, there are a variety of objectives that should be met. These include:

- Creating software that takes in an audio input consisting of a single instrument playing a range of chords
- Apply an algorithm to determine the chords found in the audio input
- Generate and apply a tune that will compliment with the characteristics found above
- Carry out a survey to discover if users believe the generated tune compliments the original chords

2. LITERATURE REVIEW

Here literature which is relevant to this project will be referred to. I will begin with a broad overview of existing software that provides augmented creativity and then look at the specifics of methods of achieving this.

2.1. Existing Software

A small amount of software is currently available which uses augmented creativity within it. These are often closed source and do not provide much information on how they have achieved their goals. However, there have been a few open source projects or purely research-based projects that will better aid in this project.

2.1.1. RockSmith

One of the current pieces of software available which provides an insight into augmented creativity includes the computer game 'RockSmith' (Ubisoft Entertainment, 2013). In this software is the presence of a 'session mode', which allows for the real-time computation of a backing track to accompany a lead guitar tune that a user plays in. This backing track is multi-instrument and consists of many desirable features, such as the ability to speed up dependant on the speed of the audio input. A visual scale is also provided to the user, showing notes that would sound pleasant with the backing track, but also updating should the user venture outside of these suggested notes to use a different scale.

Unfortunately, likely due to the type of application that RockSmith is, much of the technical justifications for how RockSmith works are kept private to the developers that created the software. For this reason, we can only speculate about how these types of artificial intelligence are created for this augmented creativity application.

2.1.2. SongSmith

SongSmith is another tool which provides a multi-instrument backing track to a single audio input. However, instead of the input being created via a guitar, this uses a vocal input (Microsoft Corporation, 2009).

Unlike RockSmith, SongSmith have dedicated research papers to explaining how their software achieves its goals. These papers describe how the frequencies of the vocal input is detected by using a pitch detection algorithm in the time domain: Autocorrelation. The specifics of how Autocorrelation works will be explored in Section 2.2.2.

The music chords that are played via SongSmith are also programmatically decided upon using a custom designed algorithm. This algorithm works by looking at the probability of a chord changing to another chord, such as a C Major to an E Major, along with the chords which are closest to a sequence of notes that have been played (Microsoft Corporation, 2009). From here the ideal chord can be chosen.

2.2. Pitch and/or Chord Detection

One of the initial aims that must be met in regards to a music based project is the ability to detect the pitch, or chords, that are present in a short sample of audio. This is the baseline for most augmented creativity projects, as processing this initial sound input allows for complementary sounds to be added to the audio – ideally resulting in an overall pleasant sounding piece of music.

2.2.1. Chromagram/Pitch Class Profile

A Chromagram, sometimes known as a Chroma Feature or Pitch Class Profile, is used as a display for the measure of intensity of the 12 different pitches (consisting of C, C#, D, D#, E, F, F#, G, G#, A, A#, B) (Jiang, 2011). In theory, this can be as simple as being a single black mark denoting full intensity of a specific pitch on a Chromagram (see section B of figure 1). However, in practice, when used on real data this will be a heavily shaded chart which can display multiple intensive frequencies at once, as shown in section D of figure 1.

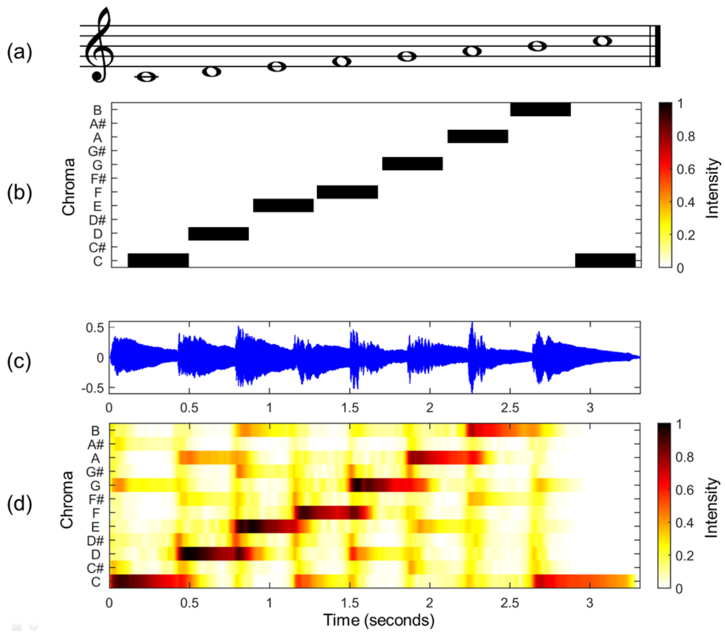

Figure 1 - An example of a Chroma Feature. Part A is the 12 pitches denoted in the score of a C Major Scale, Part B is these notes denoted in a Chroma. Part C looks at a natural audio input made from playing the C Major scale on a piano and Part D is this input in a Chroma. (Jiang, 2011)

Chromagrams are used heavily throughout the different types of chord detection, as they provide any easy storage for all of the different pitches that are present in a sample of music. Once stored, the most dominant pitches can be detected, allowing for the chords used to also be spotted.

2.2.2 Autocorrelation

Autocorrelation can be used as a pitch detection algorithm that is predominately carried out within the Time Domain (Gerolimetto, 2010). As mentioned in section 2.1.2., it is the method of choice for the SongSmith software (Microsoft Corporation, 2009).

Autocorrelation is able to find the dominant frequency in an audio input by taking the value of a signal at time t, and comparing this signal with a delayed version of itself $t + \tau$ (B. H. Suits, 2015). Doing this allows for a measure of self-similarity (MIT OpenCourseWare, 2008), allowing for the dominant frequency within the timeframe of the sample given to be produced.

In terms of songs, a finite duration waveform would be present, meaning that the waveform would only exist where $t_1 \leq t \leq t_2$. In order to then calculate the Autocorrelation of a sample, the values of the signal at t and $t + \tau$, where τ is the lag, would be multiplied and then averaged (by dividing by the number of signal samples used). The full calculation for this is shown in Figure 2.

$$\rho_{ff}(\tau) = \int_{t_1}^{t_2} f(t) f(t+\tau)\, \mathrm{d}t$$

Figure 2 - Calculation for Autocorrelation in a Finite Waveform (D.Rowell, 2008)

As is expected, this process can be repeated for as many time samples as required, giving a more accurate answer for an increase in samples used, providing that the pitch of the sample input is somewhat constant.

In terms of using autocorrelation for chord detection, the process works much the same: The same autocorrelation calculations are applied, but rather than only the dominant frequency being looked at, the frequency spectrum is used. This spectrum is often stored inside the Chromagram, and then the chords are found to be made up of the frequencies with the most presence.

One study of chord detection with Autocorrelation took this further and looked at the use of key detection to improve the chord detection rate (Zenz, 2007). By determining the key that the song was in, this allowed software to refine the set of possible chords, and these refined chords were then used as possible chords to check against with the Chromagram. Through the use of this key detection, the accuracy of chord detection through Autocorrelation went from

37% up to 50%. This accuracy level was further increased to 65% when Beat tracking (where the data was split on each beat, as chords were expected to last for a minimum of one beat) and Sequence smoothing (where the chord is rated dependant on a single chord with the fewer chord changes, and the highest rated chord was the most likely) were also applied.

Unfortunately, one of the flaws with Chord Detection is that it is susceptible to errors, due to the way that the sinusoidal waves for identical notes (in other octaves) work. If we take the note C for example, and look at the frequency and wavelength for middle C (C_4), the frequency is 261.63Hz and the wavelength is 131.87cm (B. H. Suits, Physics Department, Michigan Technological University, 2015). Going up an octave will simply just double the frequency and half the wavelength. This can be seen below in Figure 3.

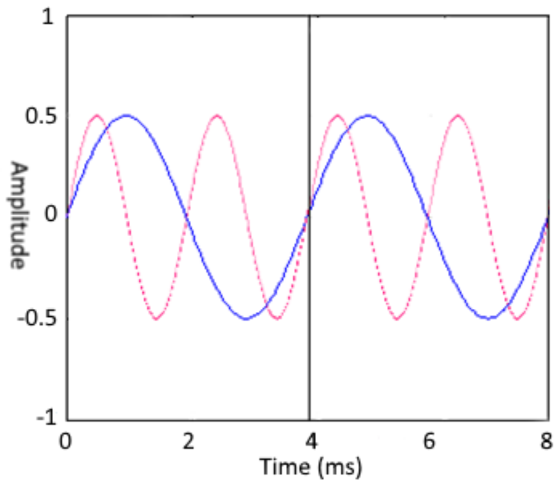

Figure 3 - The C_4 Wave in blue and the C_5 Wave in pink. For every single wave completion in C_4 there are two wave completions in C_5

These values can cause issues during the autocorrelation, where the correct note in the incorrect octave is given – a false detection. Due to the chances of this false detection occurring, autocorrelation is a technique best used in vocal implementations, as these tend to remain at a mid to low frequency where pitch range is more limited (Stanford University, 2009).

2.2.3. Fourier Transforms

Fourier Transforms are used heavily throughout the fields of mathematics and engineering, especially in audio/signal engineering (Associated Universities, Inc., 2009). They are functions, consisting of a Forward Fourier Transform and an Inverse Fourier Transform, that allow for the transformation between the time and frequency domain. When in the time domain, only the time and the amplitude information is available and is shown as a series of sinusoids. However, often this does not provide enough information for enhancing the pre-existing audio.

By using a Forward Fourier Transform, signals from the time domain can be converted to the frequency domain. Once in this frequency domain, the signal is seen a summation of the sinusoids, and the frequency content of the signal can then be determined.

Fourier Transforms are a generalisation of the Fourier series, which could only be used on periodic signals. Fourier Transforms now allow for aperiodic signals to be transformed, and for use in songs, a Discrete Fourier Transform (DFT) would be used. This is because a DFT is used for a finite numbers of samples, which would be present in a single song, rather than the theoretical Fourier Transform which uses infinite samples (Associated Universities, Inc., 2009). The Fast Fourier Transform (FFT) is an algorithm often used to compute the Discrete Fourier transform (DFT).

Prior to applying the FFT to an audio sample, the sample must be correctly set up. This requires making the sample periodic, which means to repeat the sample over the time axis. Once this is done, the sample must also be windowed. This is due to the structure of the sinusoids that will be passed into the FFT. The process of windowing will be further explained in Section 2.3., however, in its most basic form, it can be thought of as reducing the presence of the start and end of a sample so that when samples are repeated there are no harsh changes between them.

Recently research has been done to prove that the use of FFT can produce Chromagrams that allow for chord detection with accuracy levels of around 90-100% depending on the chord quality, such as Major or Diminished (Stark, 2009). This is achieved through using the FFT and then using the square root of the magnitude spectrum, which is the size of the different frequency bins. The square root of the spectrum is used in order to reduce the amplitude difference between harmonic peaks, which allows for the frequencies to be detected within a set range, rather than any range. By doing this, the noise from outside of the range does not disturb the results. From here, the chord can be detected by finding the minimum dot product between the Chromagram and a complementary bit mask of possible chords.

2.3. Window Function

A window function refers to the process of taking a signal and making the values outside of a specific range zero and bringing the values inside of the range down to 0 in a smoother way (Weisstein, n.d.). It is needed when processing audio that requires a sample to be repeated, such as was described for Fourier transforms in section 2.3.3. This is because if the sample starts at an amplitude of 0, and ends of an amplitude of 0.5, when the sample is repeated it will become disjointed due to the sinusoids not joining together. An example of this is shown in Figure 4 below. This would affect the values produced when audio processing is done on the sample, as it is not representative of the actual audio.

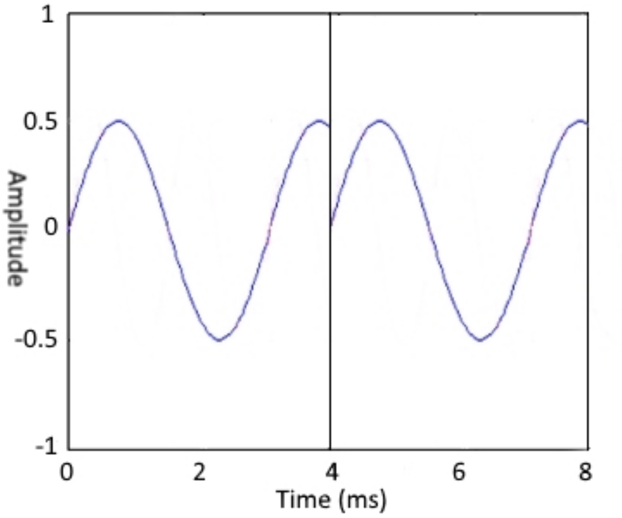

Figure 4 - An example of the repeated 4ms waveforms before windowing has been applied. Note how at the 4ms mark the waveforms do not meet.

There are many types of windowing functions that can be used, and they revolve around specific shapes. The most common type of windowing function follows the shape of a bell curve, though triangle shapes may also be used. The differences in shapes allows for different goals, such as a more accurate amplitude, as with the Hanning and Hamming windows, or lower noise floor (Anon., 2012).

2.4. Finding Complementary Notes

In order to write a pleasant sounding melody to a song, the notes used in the melody must complement chords that are being used. There are multiple different ways that these complementary notes can be found.

One way of finding complementary notes is to use a combination of the notes that are in the chord that is being played (Rockin Cowboy, 2015). This could be any combination of either C, E, or G, if the chord C major was played, for example. This could also be done in an arpeggio, in which the notes are played in a rising or falling order. However, occasionally the arpeggio should be stepped out of to stop the music becoming boring and predictable (Ewer, 2010).

Complementary notes can also be found by looking at the key that the chord sequence is in. The complementary notes are notes which are in the key that the overall chord sequence is in, and also the key of the specific chord (Rockin Cowboy, 2015). For example, when a song is in the key of C, and the chord F major is played, the melody should consist of the notes F, G, A, C, D, B, because these notes are common between the two keys. However, when playing these notes, around 50% or more of the notes in the melody should come from the notes in the chord. This ensures that the chords match the melody and it does not become disjointed.

2.5. Audio File Types

In order for software to be able to analyse music, there must be some type of audio input. Most often, songs come in the form on an MP3 file (NCH Software, n.d.), however WAV is also a standard file format used for audio.

2.5.1. WAV

The WAV file is one of the simplest file formats in terms of how the data is stored. Its data can be stored in an uncompressed PCM format, which allows for the storage of all data, with no loss of quality (Joe, 2013). However, due to this uncompressed format, the file sizes are much larger than compressed alternatives. WAV files can have compression within them (Wireman, 2013), however, this complicates and lengthens the processing of the file as it must be uncompressed prior to processing it (Reas, 2014).

The format of a WAV file is dependent on implementation. There is always a 'RIFF' descriptor chunk, and a 'fmt' sub-chunk, and a 'data' sub-chunk (Sapp, 2005). However, there can also be a 'fact' chunk in compressed versions of WAV files (Kabal, 2015).

The descriptor chunk is always made up of 3 segments, the Chunk ID, the Chunk Size and the Format, where the Chunk ID is "RIFF" and the Format is "WAVE". The Chunk Size is the size of the file. This chunk relies on the presence of the 'fmt' and 'data' sub-chunk.

The "fmt" sub-chunk is also made up of an ID and a Size, but contains the Audio Format, Number of Channels, Sample Rate, Byte Rate, Block Align, and Bits per Sample. While the data chunk is also made up of the ID and Size, with the remaining size of the data sub-chunk being the audio data. This is the raw sound data that is ready to be used for processing.

2.5.2. MP3/MP4

The other common audio file format, MP3, is most often used for streaming audio. This is due to its much smaller size when compared to a WAV file. It is smaller in size as it is a compressed format, however this is done in a lossy manor, with lossy meaning that some of the data is lost during the compression to MP3. A lossless format would lend itself better to an application with a focus on chord detection algorithms, as the data should be as unedited and as complete as possible.

2.6. Creating Audio

Another aspect that must be involved in the creation of augmented creativity applications is the actual creation of the audio which played the computationally generated notes. This can be done in a small variety of ways, but the main two that are used consist of sampling or the use of ADSR envelopes.

2.6.1. Sampling

Sampling refers to the concept of having a pre-set list files that each consist of a single note played by an instrument. These files are then joined in order to make one continuous tune. Using sampling ensures that the tone of the audio produced is as you would expect a real instrument to sound, unlike some other methods. However, due to the number of files that would be required to represent all of the possible notes, this would quickly use up space. Similarly, the quality of audio available for free that would fit this purpose is very poor, with inconsistent timing of the instrument being played being very common. The alternatives to this include expensive libraries that have been produced for this exact purpose.

2.6.2. Computationally Generated Sounds

The alternative to sampling would be to produce the audio entirely computationally. This would require finding the frequencies of each note to be used, such as the frequency for C_4 which is 261.63Hz, and then creating a sinusoidal wave of this frequency. This would be the wave for the constant sound of C_4. The sinusoidal wave can be calculated using the calculation below:

$$\text{Amplitude} * \text{Sin}((2\pi * n * \text{Frequency}) / \text{SampleRate})$$ (Goodwin, 2011)

In this calculation amplitude is the required amplitude of the sinewave (often 1), Frequency is the frequency of the soundwave, which would be 261.63 for C_4, n is the number ascending from 0 to the number of samples required, and SampleRate is the sample rate required, which is often 44,100.

In order to make this generated sound appear to be more realistic, many techniques can be applied to it. These include the use of an ADSR Envelope, or the use of the Karplus-Strong Algorithm.

The ADSR Envelope is applied to a sinusoid to help the mechanical sounding music to be edited into a more realistic style. This is done through Attack, Decay, Sustain, and Release. The attack stage is used at the start of the audio and takes the amplitude from 0 up full amplitude, and this speed is dependent on the value given to Attack, with 1 being an instant increase to full volume. The Decay is the amount of time it takes for the sound to reduce to the sustain level of amplitude. From here, the Release can be called, which takes the sound back down to an amplitude of zero.

The ADSR Envelope can have a range of different values for each section, and these values lend themselves towards certain instruments. For example, percussion instruments would have a sharp attack (zircon, 2009), as would a piano. However, a piano would have no sustain, but a long decay (Zephyr, 2009).

The Karplus-Strong Algorithm is used to create a plucking sound, used in instruments such as guitars. It is able to achieve this sound by modifying a sinusoid to reflect the vibrations that occur during the plucking motion of a real instrument (Burk, n.d.).

The Algorithm works by taking the periodic sinusoid of a specific frequency and cycling through the values within. For each value that is reached, the value is averaged with the previous value, which essentially works as a low-pass filter. The process continues, until eventually the values have been averaged so many times that the wave becomes flat, giving an amplitude of 0 (no sound). This effectively mimics a string instrument, as overtime the higher frequencies of both a string instrument and this implementation are lost.

3. RESEARCH METHODOLOGY

3.1. Overview

Throughout this section the details of the implementation will be discussed. This includes the process of inputting audio of a WAV file type, detecting the chords used via FFT and a Chromagram and Chord Detection library, composing a melody using these chords, and then writing to a second WAV file with the newly created melody. The newly created melody will consist of a tune created in the time domain, and making use of an ADSR envelope to increase the realism of the computationally generated sounds.

Due to the availability of libraries, the project was completed using C++. This is because C++ has an abundancy of Fourier related libraries, and also boasts a chord detection library with over 90% accuracy that was mentioned in Section 2.2.2. As the project looks at if chord detection algorithms can be accurate enough for a pleasant sounding melody to be generated, this would be the ideal library to use.

As this project merges concepts that have previously not been used together, it would be logical to assume that some parts of the planned implementation will not work as well as expected. For this reason, the implementation will followed an agile methodology. This flexibility allowed for changes to be made should the first method implemented not work as expected.

3.2. Audio Input

The audio input used for the project was the WAV file format. This is because it is the simplest to implement, as it does not have to be compressed, unlike MP3 formats. It is also extremely easy to find sample clips to use online. There was a need for high quality files, such as uncompressed or lossless compressed files, to be used within this project, as these retain the information needed to get the highest accuracy possible during processing of the chords.

The Audio files used within the project all originated from the same Website: Looperman (Future Web Services, 2017). They consisted of different musical styles and instruments, allowing for the possibility that a certain style of music or instrument will produce different results in the experiment. These included an electronic style upbeat sound clip (MistaTungTwista, 2017), a singular rhythm guitar sound clip (zacwilkins, 2017), two sound clips with a series of chords played on the piano, and a sound clip made up on synthesized notes (gelomelo, 2017a) (gelomelo, 2017b).

In order to be able to process the audio input, the audio was first split into frames. Once in a frame, a small section of audio could be passed into a FFT for conversion into the frequency domain.

As the application is looking at the detection of chords rather than individual notes, we will assume that a chord will be present for a minimum of 1 second, unlike notes which can change much more frequently. With this assumption in mind, it would be logical to take 1 second worth of audio to analyse at a time. The size of the frames required for each second of input can be determined by using the data taken from the WAV headers.

Within the format chunk of the WAV file there is a sampling rate that is given. This is directly related to the number of frames per second of audio. For this reason, the sample rate will be used as the number of frames to pass into the FFT at one time. Common values for this include 44,100Hz and 8,000Hz, which would mean that frame sizes such as 44,100 or 8000 would be used. In the case of the files used throughout this project, the sample rate for all files was 44,100. For this reason, we will assume a sample rate of 44,100 will always be used, and will use this reference accordingly. This is a limitation of this implementation.

In order to fill the specified frame with audio data from the WAV file, the file is first read from the data offset. However, another limitation of this project is that it only deals with wav files that store their data as 16 bit. Unfortunately, the Chromagram and Chord Detector library requires frames of a type double, so the 16 bit segments must be converted to a double. This is done by reading the first 44,100 bytes, and then cycling through them 2 bytes (16 bits) at a time.

3.3. Detecting the chords

Once the frame is filled with data from the Wav, the data must then be windowed. This is to reduce the effects of disjointed signals at the edge of the data to be sent, as once these signals are repeated, the data would have harsh sounds with each repeat. Due to the library that is being used for the detection of chords, this is required to be a Hamming window (Stark, 2009). The hamming window is applied to each frame of data, which prepares them to be passed through a Fourier Transform.

The Hamming window that is used creates a bell curve on the audio sample, where the start and end of the sample have reduced amplitude, but it is not set to 0 (National Instruments Corporation, 2017). This is an enhancement of the Hanning window, which ensured the amplitude at the beginning and end were set to 0.

Upon completion of the windowing, the frames are then passed into a Fourier Transform, using the Chromagram and Chord Detection Library (Stark, 2009), where they are transformed from the time domain and into the frequency domain. This allows the frequencies of the 12 pitches to be detected and compared. These frequencies are plotted in a Chromagram, and then can be used for the Chord Detection.

The Chord detection is then applied by passing the Chromagram into the Chord Detection section of the Chromagram and Chord Detection library (Stark, 2009). From here, the library finds the dot product with the values in the Chromagram against all of the complementary bit masks of all the possible chords. The correct chord gives the lowest value here. This library was chosen as it boasts a higher accuracy rating over other chord detection methods, as it included chords such as Diminished 7th, instead of just the standard major minor chords that are in other chord detection algorithms.

3.4. Creating the melody

As the chords have now been detected, the melody can be generated. Due to the overall aim of this project being related to the possibility of creating a melody that relies on the accuracy of the chord detection, the melody created should be very simple. This will limit the effects of a melody that has been poorly created affecting the results.

For this reason, the melody generated follows a very simple guideline. A list of possible notes is created, which contains the 3 notes which are in the chord which has been detected. If a more advanced implementation were to be created, this could include notes that were also in the key of the song and the key of the chord, such as was explained in Section 2.4. However, this would be better for future implementations.

The starting note and ending note within a chord progression was required to always be the root note of the chord. Whilst inside of this chord progression a random number generator decide which note would be used, choosing from the list created. A random number generator also chose the length of each note to play, ranging from 3 to 7 seconds long for the very first note, and then up to 10 seconds long for any additional notes within the chord progression – but never exceeding the length of the chord progression.

The actual sound of the melody is created by first generating a Sine wave, as explained in Section 2.6.2, using the calculation sin((2 * pi * n * frequency) / sampling rate), where the frequency is the frequency of the note being created, and n is the value 0 incrementing until it hits the total number of samples required, where there is 44100 per second.

After the wave has been created an ADSR envelope is used to make it sound less mechanical. The ADSR envelope was chosen rather than the sampling, due to the poor availability of good samples. Also, the ADSR envelope could be adjusted to give different styles of instrument. The ADSR Envelope used came from a library (Redmon, 2012), which just required the values wanted for each stage in the envelope. The overall sound wanted from this audio was a flute style. This requires a fast attack and decay, with a slow release. For this reason, the values used for the ADSR were 0.7, 0.7, 0.4 and 8 respectively. This was decided after much configuration, to find the ideal values.

3.5. Creating the Complementary WAV file

In order to play the generated melody to users, it must be present in some type of file. With the WAV file format already being used for the original audio data, it is logical to use the WAV file again for the melody audio data. Currently, the project will always generate a new WAV file for the melody, as it cannot overwrite the existing WAV file so that the new melody and existing chords can be played together.

In order to create the new WAV file, a new file is created and given the extension of '.wav'. From here, the required headers mentioned in section 2.5.1 are added. In order to keep the standards set by the original WAV files, this new file is set up to have a sample rate of 44100, and uses two channels. Once the headers have been filled in, the audio data can be written to the wav file. Only after this audio has been written to file can the headers be updated to specify the size of the data chunk.

3.6. Combining the original audio and newly generated audio

Due to limitations in the software written, the newly generated audio was written in a separate WAV file to the original audio which contained the chords. As the project looks at creating a pleasant sounding melody which compliments the chords detected from the original audio file, these two WAV files would either have to be played simultaneously, or otherwise combined.

Since playing two separate files simultaneously is quite hard to do using just standard audio players, it makes sense to use pre-existing software to combine the two files. This would ensure that they are definitely in time with each other, as there would be no place for human error.

In order to combine the two files the software Lexis Audio Editor (pamsys, 2017) was used. In this, the original audio file was opened, and then the new audio file was imported and mixed. This was then saved, and used as the WAV file that users would listen to as the completed product.

3.7. Collecting the project data

The overall aim of the project was subjective, as it is dependent on a person's opinion of if the music generated sounded pleasant. This left a requirement to gather data from volunteers, to either prove or disprove the aim. In order to do this, a questionnaire was written, questionnaire 1. However, due to reasons mentioned in Section 4.1, this questionnaire did not provide sufficient information to either prove or disprove the aim. For this reason, a second survey was written, questionnaire 2. The process used in creating and completing the surveys will be described below.

3.7.1. Questionnaire 1 – The General

The initial questionnaire created for this project attempted to find if people agreed that the chord detection algorithms were accurate enough to allow for the creation of a complementary melody. However, due to the reasons explained in section 4.1, the results from this questionnaire were inconclusive. This lead to the creation of a second questionnaire.

This first questionnaire was distributed rather widely, being posted on multiple social media websites, by multiple people. Overall, the questionnaire was shared by 4 different people, allowing it to reach a potential 800 people, from local areas to other countries, including Spain, and Canada. However, the sample selection were a non-probability sample, as only people with specific social media connections would have had access to the questionnaire. There were also no limitations on the type of person that was able to complete the questionnaire.

The questionnaire was made up of 15 questions, the first 3 of which focused on the volunteer's musical knowledge. These questions were included to detect if volunteers with higher or lower levels of music knowledge tended towards different results when analysing the audio, as it may be that volunteers with more musical knowledge may pick up on more flaws.

The initial question consisted of a Likert scale question, allowing the volunteer to rate what their perceived knowledge of the technical side of music is, and then a further two questions were asked to determine if the volunteer had any qualifications related to music, and if so, what these were. They were advised that qualifications such as music theory were to be included.

The remaining 12 questions were split into sets of 4, where each set contained 3 questions. In order to answer these questions, the volunteer first had to listen to a sample section of music that was created using the methods explained above. The sample audio supplied were all between 29 and 48 seconds, so not to lose the interest of the volunteers. The sample audio had to be at least the minimum length given, as this ensured that enough chord changes had occurred, allowing for the chord detection algorithms to be used a substantial amount.

The set of 3 questions were written to determine if the participant felt as though there was something incorrect about the song, such as that the added melody used notes which did not fit in with the overall tune of the song. In this instance, it would be assumed that the chord detection software's inaccuracy had been obvious to the participant. The aim of these 3 questions were also to discover if the participant liked the music with the addition of the computational generated melody.

These aims were reached by using another Likert scale question, asking the user to rate how much they liked the audio clip that was played. They were then asked a dichotomous question looking at if the song seemed technically correct. Finally, they were asked a qualitative question asking for their reasoning behind the answers to the first two questions. Answers to this question were expected to determine if the volunteers opinion of the music was related to the audio added, or if it was simply a style of music that they did not enjoy. The answers to this question were also expected to explain which areas of the song they thought did not sound technically correct. This was supposed to determine if this was related to the tune produced from the chord detection algorithms, and if it was due to an inaccuracy, rather than the original tune seeming to have inaccuracies.

3.7.2. Questionnaire 2 – The Specifics

Questionnaire 2 was written to narrow down the questions from questionnaire 1, in order to be more specific in proving the hypothesis of the project. This was achieved by asking questions that were more specific to the individual features of the audio played, rather than expecting the participant to point out specific features.

The second questionnaire shared similar characteristics of the original questionnaire in terms of volunteers. This is because the questionnaire was, again, shared via social media websites, by 4 people. This allowed it to be accessed by over 1500 people, and the survey was also shared by word of mouth. Once again, this was non-probability sample.

This questionnaire was made up of 6 questions this time. The reduction in question numbers was an attempt to get a higher rate of completion of the survey, as boredom from too many questions could have been the reason for a low completion rate with questionnaire 1.

There was also a reduction in the audio clips that were used in this questionnaire. This time only two clips were used. Again, this was done to increase the completion rate, by reducing the amount of time that the questionnaire takes.

The 6 questions were made up of 2 sets of identical questions. The 3 questions were similar to questionnaire 1, as they all related to an audio clip that was produced via the methodology above. In order to narrow down the answers from questionnaire 1, the questions were all dichotomous here, consisting of only Yes or No answers. The three answers looked at the most common responses received from questionnaire 1: If the participant thought the melody was out of sync with the rest of the song, if the melody had any clashing notes with the rest of the song, and if they liked the overall sounded of the instrument from the melody. These questions served the purpose of explaining the reasoning behind while some participants believed the audio clip may have been technically incorrect. The main question that we are interested in here is question 2. This is because question 2 is directly related to the accuracy of the chord detection algorithm used. Questions 1 and 3 are instead related to the implementation used to create the melody, which may have been unsuccessful, but not related the accuracy of the chord detection.

4. FINDINGS AND ANALYSIS

This section will consider the results from both questionnaire 1 and 2 and how these results can be applied to either disprove or prove the aim. The results will be analysed in terms of sample size, sample quality, and answers given.

4.1. Findings from Questionnaire 1

As mentioned in Section 3.7.1, the sample for Questionnaire 1 was gathered through the use of social media. Due to this type of sample gathering, there is a risk that the sample will not be generalizable to the public. However, due to this being shared in a public manor, by multiple people from different areas, this should reduce the risk.

Another flaw within the sample used in the size of the sample. The sample is rather small, having only 17 people complete it. This small sample size would increase the standard deviation of the results, meaning that they are less accurate. However, the results will be considered while taking these flaws into consideration.

The findings from Questionnaire 1 were vaguer than ideal responses. Taking a look at Appendix 1 (Section 7.1) for example, it is seen that many of the explanations given are very basic and keeping to one line answers. Some answers are also very simplistic, simply stating that the music "Sounds good to me", and "feels more real".

4.1.1. Audio Sample 1

The results from Questionnaire 1 can all be found in Appendix 1. The questions that related to Audio Sample 1 include question 4, 5 and 6. Refer to this appendix for any clarification required.

In question 4, the participants were first asked to rate how much they enjoyed the music. This question would not give us the information needed regarding the chord detection algorithm by itself, as it could be that the user just dislikes a certain style of music that was played. However, in combination with the following two questions, which consider if the volunteer found the musical technically correct, and why, it should allow the understanding required to see if their dislike was due to a technical flaw.

The users' opinion of the first track was very varied, ranging from a score of 1, being strongly dislike, to a score of 9, where 10 is strongly like. This gave a mean score of 5, which can be seen as feeling indifferent about the audio. Whilst not necessarily being bad, this result could really only be interpreted in terms of if our aim was met by considering these results with the following two questions results.

Looking at the second question related to sample 1, question 5, this split view seems to continue, with only 59% thinking that the music sounds technically correct, while the other 41% felt that the music was incorrect technically. Looking at these percentages alone, it appears as though the aim is starting to become disproved, as the technical processing of the song (e.g. the chord detection algorithm) was too poor to allow for the creation of a pleasant sounding song.

However, the explanation of these answers comes from part 3 of Audio Sample 1, question 6. Here it is seen that there are varied responses as to why the volunteers found that the song was not technically correct, and why they gave a low score when rating their opinion of the audio tune.

Responses to this question included comments such as "The tune doesn't match with the backing. The tone matches but the timing does not", and "I didn't like the sound of the wind instrument being used".

In this situation, it appears that the answers to question 5 could have been giving a false negative, in which the results were showing that the music produced was technically incorrect (and this was being interpreted to believe that this would have been due to the incorrect chords given from the detection algorithm). However, the music was often rated to be technically incorrect due to another cause.

Looking at the answers to the final question, there were no comments that specifically commented on the pitch of the generated notes clashing against what was already present. There were 2 comments that could be interpreted to be a clashing of the notes, such as "Lack of dischord", and "A few of the notes seemed like randomn blips". However, the majority were related to the "odd warbling sound" and "mechanical" sounds, which would have been caused by the use of the ADSR envelope, or that "The added tones were out of sync with the rest of the piece" and "The tone matches but the timing does not" which would have been related to the actual rhythm in the melody. As neither of these complaints would have been caused by the chord detection algorithm, as this only provided the information to do with the pitch of the notes used, these technical failures cannot be pinned down to the accuracy of the chord detection algorithm.

4.1.2. Audio Sample 2

The results discussed in this section can be found in Appendix 1, and refers to questions 7-9.

Question 7 related to Audio Sample 2 received slightly higher ratings of how much the participant liked that music that Audio Sample 1. No participants gave a score of 1, which is strongly dislike, while 2 participants gave a score of 9. The ratings were not substantially different however, as the mean still averaged at 5, when rounded down. Again, this was a very average score, which did not shows an indifference to the music.

The number of users that rated this sample as being technically correct were also exactly the same as the previous audio sample. However, when looking at question 9, there were some specific comments where the volunteers felt as though “It sounds like there was an error half way through the song, as the note played clashes with what is already there”, However this was only 2 of the participants that felt this way.

4.1.3. Audio Sample 3

The results discussed in this section can be found in Appendix 1, and refers to questions 10-12.

As in previous examples, Audio Sample 3 had an average rating of 5, when asked if the participant liked the song. However, unlike the previous samples, this has the highest rate of being seen as technically correct – with 71% of participants thinking the song was technically correct.

The trend continues into Sample 3, with the comments on why the song was not technically correct referring to the effect that the ADSR had on the song such as “The song crackled in parts” or the actual rhythm of the melody with “The added tones were out of sync”. However, this time not a single comment was related to, or could be related to, the pitch of the melody notes. This would assume that the chord detection algorithm was accurate enough to allow for the creation of a melody – however the melody creation had a couple of issues with its creation.

4.1.4. Audio Sample 4

The results discussed in this section can be found in Appendix 1, and refers to questions 13-15.

Audio Sample 4 was the most successful sample that was created. This achieved scores of between 5 and 9 when asked how much people liked the audio, and gave a mean of around 7. With an average score of 7, it is assumed that not only did people like the style of music of the original song, but that there were also no noticeable flaws in this song that caused volunteers to dislike it.

This can be proved further by looking at the results from question 13. This question looked at if the song was assumed to be technically correct, and just over 88% of volunteers said that the song was technically correct. Again the comments were related to the ADSR envelope creating a “dull and mechanical” sound, or the “to the slightly late/delayed” parts of the song.

4.1.5 Entire Questionnaire Overview

As mentioned when first discussing the questionnaire, the starting 3 questions are related to the volunteer’s musical knowledge. This was to attempt to find if there were a difference in results between those with less musical knowledge and those with more. For this reason, the people that scored themselves as a 5 or more for the first question “On a scale of 1 to 10, with 1 being having no knowledge, and 10 being extremely knowledgeable, how much knowledge of the technical side of music do you think you have?” are treated as the knowledgeable group, whilst those who scored less than 5 are seen as the more unknowledgeable group. The results given by people in the knowledgeable group can be send in Appendix 2.

It was expected that the volunteers that had the most musical knowledge would be the most critical of the audio that is played. However, these volunteers actually gave the results that were most likely to prove the aim. They gave higher than average scores for liking the music, with the lowest mean being a 6 and the highest mean being a 7.2, and also were much likely to find the audio technically correct, with some samples hitting 100% thinking it was technically correct, and the lowest score still being as high as 80%. Unfortunately, this sample size is only 5 participants, so may not be representative of the larger group of knowledgeable musicians. However, it is a very positive start, and goes a long way in proving that the melody created using information from the chord detection algorithms can be accurate enough to sound pleasant, and have no noticeable flaws.

However, excluding this set of positive results, there were a distinct lack of comments that were related to the pitch of the notes in the melody. As this is the only aspect that is taken from the values produced in the chord detection algorithm, these comments are the only ones that will clearly disprove the aim.

Unfortunately, the lack of comments about the pitch being incorrect does not prove that the pitch was in fact correct. It just shows that none of the participants thought it was necessary to include. For this reason, it was felt that the second questionnaire was needed, and this questionnaire would specifically ask the volunteer questions related to the pitch, the overall sound of the instrument, and the timing of the notes.

It is also worth mentioning that the overall response rate for questionnaire 1 was rather low, with 36 people starting the questionnaire, but only 17 participants finishing. This gave a response rate of only 47%. A large proportion of people (50%) stopped the questionnaire upon reaching the section in which music was played. This suggests that the volunteers decided not to continue due to the expectation to listen to the music, and it may have been that they were not aware this was a requirement (though this was stated in the Participant Briefing section). As a second questionnaire will be created, this point will be highlighted prior to accessing the survey, such as when the link to the questionnaire is shared with people and also having a page dedicated to ensuring the user is in a quiet place and is able to listen to the audio. This should ensure a higher response rate for the survey, and hopefully provide a larger set of data to analyse.

4.2. Findings from Questionnaire 2

The purpose of questionnaire 2 was the determine if volunteers felt that notes clashed between the melody, which was computationally created using data from chord detection algorithms, and the original audio. It is to be used to supplement the results found in questionnaire 1. The results for Questionnaire 2 can be found in Appendix 3, and the questions 1-3 are related to audio sample 1, while questions 4-6 are related to Audio Sample 2.

Questionnaire 2 has the same flaws present as in questionnaire 1, due to the sample gathering. The small sample size of 17 means that just a couple of responses outside of the norm will skew the data one way. Unfortunately, this flaw cannot be rectified without ta larger sample sized being used.

When asked more specific questions about the audio samples, the results became much clearer than in Questionnaire 1. Within audio sample 1, 82% of people said that there were no clashing notes that were played in the sample. This combined with the standard error value of 0.1 (meaning 10% chance of error when converting to general population, due to the sample size used), gives a very high likelihood that the use of chord detection algorithms was accurate enough for the information it supplied to be used in melody creation.

Similarly, the somewhat high score of 41% of people who did not like the sound of the instrument that was added, shows even more that the level of dislike in the audio played was due to either the original music not being to their liking, receiving comments from questionnaire

1 such as “isn't really my type of music”, or due to flaws in the implementation of the melody, such as the use of the ADSR creating dull sounds.

The results were similar for audio sample 2, this time with 88% of people thinking that there were no clashing notes in the audio that was produced. This sample was the only sample that the chords used for made available, and so the actual accuracy levels could be checked. The chord detection algorithm gave a total of 52 values/chords throughout the sample. Of those 52 chords, 4 chords were incorrect. However, these incorrect chords appeared during the change of chords. This would give an accuracy rating of 92%. This shows that 88% of people did not notice any flaws in the chord detection, when the accuracy was at 92%. As the chord detection algorithm used boasts a detection rate of >90% for all chord types, this shows that the algorithm is in fact accurate enough to allow for the creation of a computationally generated complementary melody, however this implementation wasn’t incredibly successful due to other factors.

5. CONCLUSIONS AND FUTURE RECOMMENDATIONS

From the study conducted it has been found that the accuracy of chord detection algorithms is high enough that the information gained from them can give the notes for a melody to be played in. Unfortunately, it has not proven that these notes can create a pleasant sounding complimentary melody, due to the low results in Questionnaire 1. This could be proven, provided that the implementation for creating the audio sounds and deciding on the timing on each note is improved. There would be no reason that an improved implementation would not allow for the complementary melody to be produced.

For this reason, future work should look at improving the implementation of the overall melody. This could include the use of sampling instead of computationally created sounds, which would provide a much more natural sound. Similarly, research should be done into the timing of melodies, to improve upon this, possibly looking into beat detection. A further aspect to look at could be increasing the range of notes that are viable for each chord, by using the algorithm such as the one described in section 2.4 that uses the key of the song. This would further test the abilities of creative ability in software, and would make the music much more interesting to hear, being more varied.

6. BIBLIOGRAPHY

Anon., 2012. *Window Functions in Spectrum Analyzers.* [Online] Available at: http://www.tek.com/blog/window-functions-spectrum-analyzers [Accessed 24 April 2017].

Associated Universities, Inc., 2009. *Fourier Transforms.* [Online] Available at: http://www.cv.nrao.edu/course/astr534/FourierTransforms.html [Accessed 23 03 2017].

B. H. Suits, Physics Department, Michigan Technological University, 2015. *Frequencies for equal-tempered scale, A4 = 440 Hz.* [Online] Available at: http://www.phy.mtu.edu/~suits/notefreqs.html [Accessed 21 03 2017].

B. H. Suits, P. D. M. T. U., 2015. *Autocorrelation (for sound signals).* [Online] Available at: http://www.phy.mtu.edu/~suits/autocorrelation.html [Accessed 03 21 2017].

Burk, n.d. *Chapter 4: The Synthesis of Sound by Computer.* [Online] Available at: http://cmc.music.columbia.edu/musicandcomputers/chapter4/04_09.php [Accessed 26 April 2017].

D.Rowell, 2008. *2.161 Signal Processing: Continuous and Discrete.* [Online] Available at: https://ocw.mit.edu/courses/mechanical-engineering/2-161-signal-processing-continuous-and-discrete-fall-2008/lecture-notes/lecture_22.pdf
[Accessed 23 03 2017].

Demiralay, O., n.d. *CMPT 468/768 Final Project.* [Online] Available at: https://www.cs.sfu.ca/~tamaras/project468_odemiral/ [Accessed 26 April 2017].

Ewer, G., 2010. *You've Got a Chord Progression- How Do You Create a Melody?.* [Online] Available at: http://www.secretsofsongwriting.com/2010/06/04/youve-got-a-chord-progression-how-do-you-create-a-melody/
[Accessed 23 03 2017].

Future Web Services, 2017. *Looperman.* [Online] Available at: https://www.looperman.com/ [Accessed 27 April 2017].

gelomelo, 2017. *can we meet again someday.* [Sound Recording].

gelomelo, 2017. *mango daze.* [Sound Recording].

Gerolimetto, M., 2010. *Autocorrelation function analysis.* [Online] Available at: http://www.dst.unive.it/~margherita/TSLectureNotes3.pdf [Accessed 23 03 2017].

Goodwin, S., 2011. *Creating a sine wave at a specific frequency and sample rate.* [Online] Available at: https://www.daniweb.com/programming/software-development/threads/354944/creating-a-sine-wave-at-a-specific-frequency-and-sample-rate [Accessed 25 April 2017].

Jiang, N., 2011. *Analyzing Chroma Feature Types for Automated Chord Recognition.* [Online] Available at: https://www.audiolabs-erlangen.de/content/05-fau/professor/00-mueller/03-publications/2011_JiangGroscheKonzMueller_ChordRecognitionEvaluation_AES42-Ilmenau.pdf
[Accessed 24 April 2017].

Joe, 2013. *How do MP3 and WAV Files Differ.* [Online] Available at: http://www.dawsons.co.uk/blog/how-do-mp3-and-wav-files-differ [Accessed 24 April 2017].

Kabal, P. P., 2015. *Audio File Format Specifications.* [Online] Available at: http://www-mmsp.ece.mcgill.ca/Documents/AudioFormats/WAVE/WAVE.html [Accessed 25 April 2017].

Microsoft Corporation, 2009. *The Research Behind SongSmith.* [Online] Available at: http://songsmith.ms/research.html [Accessed 21 03 2017].

MistaTungTwista, 2017. *128 Cmin Plucks -OPENED sidechained.* [Sound Recording].

MIT OpenCourseWare, 2008. *2.161 Signal Processing: Continuous and Discrete.* [Online] Available at: https://ocw.mit.edu/courses/mechanical-engineering/2-161-signal-processing-continuous-and-discrete-fall-2008/lecture-notes/lecture_21.pdf
[Accessed 03 21 2017].

National Instruments Corporation, 2017. *Characteristics of Different Smoothing Windows.* [Online]

Available at: http://zone.ni.com/reference/en-XX/help/371361H-01/lvanlsconcepts/char_smoothing_windows/
[Accessed 27 April 2017].

NCH Software, n.d. *Audio File Formats.* [Online] Available at: http://www.nch.com.au/acm/formats.html
[Accessed 24 April 2017].

pamsys, 2017. *Lexis Audio Editor,* s.l.: pamsys.

Reas, C., 2014. *Processing: A Programming Handbook for Visual Designers and Artists.* Second ed. s.l.:MIT Press.

Redmon, N., 2012. *Envelope generators—ADSR code.* [Online] Available at: http://www.earlevel.com/main/2013/06/03/envelope-generators-adsr-code/
[Accessed 27 April 2017].

Rockin Cowboy, 2015. *What notes to sing with chords.* [Online] Available at: https://music.stackexchange.com/questions/30580/what-notes-to-sing-with-chords
[Accessed 24 April 2017].

Rocksmith 2014 - The Fastest Way to Learn Guitar, 2014. *Rocksmith 2014 Edition - Session Mode Tutorial.* s.l.:s.n.

Sapp, C. S., 2004. *WAVE PCM soundfile format.* [Online] Available at: http://soundfile.sapp.org/doc/WaveFormat/
[Accessed 23 03 2017].

Sapp, C. S., 2005. *WAVE PCM soundfile format.* [Online] Available at: http://soundfile.sapp.org/doc/WaveFormat/
[Accessed 25 April 2017].

Sony CSL, n.d. *FlowComposer: Composing with AI.* [Online] Available at: http://www.flow-machines.com/flowcomposer-composing-with-ai/
[Accessed 20 02 2017].

Stanford University, 2009. *PITCH DETECTION METHODS REVIEW.* [Online] Available at: https://ccrma.stanford.edu/~pdelac/154/m154paper.htm
[Accessed 23 03 2917].

Stark, A. M., 2009. *REAL-TIME CHORD RECOGNITION FOR LIVE PERFORMANCE.* [Online]
Available at: http://eecs.qmul.ac.uk/~markp/2009/StarkPlumbley09-icmc.pdf
[Accessed 24 April 2017].

Ubisoft Entertainment, 2013. *GENERAL FAQ.* [Online]
Available at: https://rocksmith.ubisoft.com/rocksmith/en-us/faq/index.aspx
[Accessed 26 April 2017].

Ubisoft Entertainment, 2013. *RockSmith.* s.l.:s.n.

Weisstein, E. W., n.d. *Apodization Function.* [Online]
Available at: http://mathworld.wolfram.com/ApodizationFunction.html
[Accessed 24 April 2017].

Wireman, C., 2013. *Sound Formats and Their Uses in Games.* [Online]
Available at: https://www.gamedev.net/resources/_/technical/game-programming/sound-formats-and-their-uses-in-games-r1902
[Accessed 25 April 2017].

zacwilkins, 2017. *Minor Guitar Rhythm.* [Sound Recording].

Zenz, V., 2007. *AUTOMATIC CHORD DETECTION INCORPORATING BEAT AND KEY DETECTION.* [Online]
Available at: http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=4728534
[Accessed 24 April 2017].

Zephyr, 2009. *Common ADSR envelopes.* [Online]
Available at: http://ocremix.org/community/topic/16752-common-adsr-envelopes/
[Accessed 25 April 2017].

zircon, 2009. *Common ADSR envelopes.* [Online]
Available at: http://ocremix.org/community/topic/16752-common-adsr-envelopes/
[Accessed 25 April 2017].

7. APPENDICES

7.1. Appendix 1

Augmented Creativity Questionnaire

Question 1

On a scale of 1 to 10, with 1 being having no knowledge, and 10 being extremely knowledgeable, how much knowledge of the technical side of music do you think you have?

			Response Percent	**Response Total**
1	1		29.41%	5
2	2		23.53%	4
3	3		5.88%	1
4	4		11.76%	2
5	5		5.88%	1
6	6		0.00%	0
7	7		11.76%	2
8	8		11.76%	2
9	9		0.00%	0
10	10		0.00%	0
Analysis	Mean: 3.47	Std. Deviation: 2.52	Satisfaction Rate: 27.45	answered 17
	Variance: 6.37	Std. Error: 0.61		skipped 0

Question 2

Do you have any Grades in any instruments or music theory?				
			Response Percent	Response Total
1	Yes		23.53%	4
2	No		76.47%	13

Analysis							
Analysis	Mean:	1.76	Std. Deviation:	0.42	Satisfaction Rate:	76.47	
	Variance:	0.18	Std. Error:	0.1			

answered	17
skipped	0

Question 3

If yes, what do you have grades in, and what grades have been achieved?			
		Response Percent	Response Total
1	Open-Ended Question	100.00%	5

1	11/04/17 5:07PM ID: 55439931	Grade 2 in theory
2	11/04/17 6:19PM ID: 55442822	N/A
3	11/04/17 8:32PM ID: 55447809	1
4	11/04/17 8:44PM ID: 55448146	Grade 3 Trumpet Grade 3 Piano Grade 4 Theory
5	24/04/17 11:42AM ID: 56348925	Grade 4 theory

answered	5
skipped	12

Sample 1 is played here. The sample can be played by following this URL: https://www.youtube.com/watch?v=MD8GLRpIF8U

Question 4

On a scale of 1 to 10, with 1 being strongly dislike and 10 being strongly enjoy, how much do you like this piece of music?

			Response Percent	Response Total
1	1		11.76%	2
2	2		5.88%	1
3	3		17.65%	3
4	4		5.88%	1
5	5		5.88%	1
6	6		5.88%	1
7	7		29.41%	5
8	8		11.76%	2
9	9		5.88%	1
10	10		0.00%	0
Analysis	Mean: 5.18; Std. Deviation: 2.53; Satisfaction Rate: 46.41		answered	17
	Variance: 6.38; Std. Error: 0.61		skipped	0

Question 5

Did you think this song was technically correct? (with technically correct meaning that the sound of the song did not sound incorrect/as though an error had been made)					
				Response Percent	**Response Total**
1	Yes			58.82%	10
2	No			41.18%	7
Analysis	Mean: 1.41	Std. Deviation: 0.49	Satisfaction Rate: 41.18	answered	17
	Variance: 0.24	Std. Error: 0.12		skipped	0

Question 6

Why did you like/dislike this clip, and why did you feel this song was/was not technically correct?			
		Response Percent	**Response Total**
1	Open-Ended Question	100.00%	17

1	11/04/17 3:58PM ID: 55435956	Odd warbling noises in the background behind the regular music progression
2	11/04/17 4:05PM ID: 55436275	it sounded calming, something about it sounded unusual though
3	11/04/17 4:32PM ID: 55438150	The added tones were out of sync with the rest of the piece
4	11/04/17 5:07PM ID: 55439931	The tune doesn't match with the backing. The tone matches but the timing does not
5	11/04/17 5:25PM ID: 55440542	Sounded good to me
6	11/04/17 5:54PM ID: 55441998	thought it was rubbish and mechanical

Did you think this song was technically correct? (with technically correct meaning that the sound of the song did not sound incorrect/as though an error had been made)				
			Response Percent	**Response Total**
7	11/04/17 6:19PM ID: 55442822	I liked the top but the bottom was a bit plinky-plonky.		
8	11/04/17 8:24PM ID: 55447090	I like the peice of music because it sounded peaceful and I liked the flow of sounds. The song sounded technically correct because there was nothing that sounded too offbeat or unexpected.		
9	11/04/17 8:32PM ID: 55447809	It was an easy listening piece of music, calming. Sounds blended and flowed together.		
10	11/04/17 8:39PM ID: 55448135	Sounds like echo the dolphin		
11	11/04/17 8:44PM ID: 55448146	Movement through the guitar chord was distracting. The different instruments in the music seemed disjointed - no blend		
12	11/04/17 8:55PM ID: 55448688	A few of the notes seemed like randomn blips, but it was a very calming tune.		
13	11/04/17 9:25PM ID: 55449746	The audio flowed quite well. The audio didn't have an aggressive nature so was a pleasant listen.		
14	12/04/17 12:46AM ID: 55454756	I didn't like the sound of the wind instrument being used		
15	12/04/17 5:15AM ID: 55457960	Lack of dischord & interesting left/right balance between synthesised / plucked string sounds.		
16	24/04/17 11:42AM ID: 56348925	I liked this music because it felt like I was having a masarge and it felt relaxing. I thaught the clip was technically correct because it blended well together.		
17	24/04/17 11:56AM ID: 56351120	It's soothing and sounds nice		
			answered	17
			skipped	0

Sample 2 is played here. The sample can be played by following this URL: https://www.youtube.com/watch?v=DEXZGZ4XC2Q

Question 7

On a scale of 1 to 10, with 1 being strongly dislike and 10 being strongly enjoy, how much do you like this piece of music?				
			Response Percent	**Response Total**
1	1		0.00%	0
2	2		5.88%	1
3	3		11.76%	2
4	4		11.76%	2
5	5		29.41%	5
6	6		11.76%	2
7	7		17.65%	3
8	8		0.00%	0
9	9		11.76%	2
10	10		0.00%	0
Analysis	Mean: 5.41	Std. Deviation: 1.91 Satisfaction Rate: 49.02	answered	17
	Variance: 3.65	Std. Error: 0.46	skipped	0

Question 8

Did you think this song was technically correct? (with technically correct meaning that the sound of the song did not sound incorrect/as though an error had been made)									
								Response Percent	Response Total
1	Yes							58.82%	10
2	No							41.18%	7
Analysis	Mean:	1.41	Std. Deviation:	0.49	Satisfaction Rate:	41.18		answered	17
	Variance:	0.24	Std. Error:	0.12				skipped	0

Question 9

Why did you like/dislike this clip, and why did you feel this song was/was not technically correct?				
			Response Percent	Response Total
1	Open-Ended Question		100.00%	17
	1	11/04/17 3:58PM ID: 55435956	The warbling noises don't fit with the main foreground music and create a weird dissonance making the music cause headaches.	
	2	11/04/17 4:05PM ID: 55436275	the volume seems to vary	
	3	11/04/17 4:32PM ID: 55438150	The added tones didn't fit the piece	
	4	11/04/17 5:07PM ID: 55439931	It sounds like there was an error half way through the song, as the note played clashes with what is already there	
	5	11/04/17 5:25PM ID: 55440542	The piece of music sounded upbeat but quite jumpy in parts.	

Why did you like/dislike this clip, and why did you feel this song was/was not technically correct?				
			Response Percent	Response Total
6	11/04/17 5:54PM ID: 55441998	got a bit more life to it		
7	11/04/17 6:19PM ID: 55442822	Groovy, very 80s (no bad thing) I almost reached for my scrunchies and ankle warmers.		
8	11/04/17 8:24PM ID: 55447090	Not my style of music compared to the previous example. It sounded technically correct as everything sounded like it was following a set pattern .		
9	11/04/17 8:32PM ID: 55447809	It was a lot more upbeat which isn't really my type of music. All sounds blended and sounded like they belonged in the music.		
10	11/04/17 8:39PM ID: 55448135	More up beat		
11	11/04/17 8:44PM ID: 55448146	More cohesive than clip 1. Seemed all technically produced and the different sounds blended well together		
12	11/04/17 8:55PM ID: 55448688	Again, it felt as though there were some random notes put into the clip at odd times.		
13	11/04/17 9:25PM ID: 55449746	Same as before.		
14	12/04/17 12:46AM ID: 55454756	I didn't like the two different sounds that were occurring. Sounded technically correct as there were no identifiable mistakes.		
15	12/04/17 5:15AM ID: 55457960	Found chordal overlay tone to clash with stronger rhythym of more staccato synth		
16	24/04/17 11:42AM ID: 56348925	I liked the music because the beat made me feel like I wanted to dance. I thought the music was technically correct because nothing sounded out of place		
17	24/04/17 11:56AM ID: 56351120	Puts me in a groovy mood		
			answered	17
			skipped	0

Sample 3 is played here. The sample can be played by following this URL: https://www.youtube.com/watch?v=-9xN15aHQZk

Question 10

On a scale of 1 to 10, with 1 being strongly dislike and 10 being strongly enjoy, how much do you like this piece of music?			Response Percent	Response Total
1	1		0.00%	0
2	2		5.88%	1
3	3		11.76%	2
4	4		17.65%	3
5	5		11.76%	2
6	6		23.53%	4
7	7		17.65%	3
8	8		5.88%	1
9	9		5.88%	1
10	10		0.00%	0
Analysis	Mean: 5.41; Std. Deviation: 1.85; Satisfaction Rate: 49.02		answered	17
	Variance: 3.42; Std. Error: 0.45		skipped	0

Question 11

Did you think this song was technically correct? (with technically correct meaning that the sound of the song did not sound incorrect/as though an error had been made)				Response Percent	Response Total
1	Yes			70.59%	12
2	No			29.41%	5

Analysis								
Mean:	1.29	Std. Deviation:	0.46	Satisfaction Rate:	29.41		answered	17
Variance:	0.21	Std. Error:	0.11				skipped	0

Question 12

Why did you like/dislike this clip, and why did you feel this song was/was not technically correct?		Response Percent	Response Total
1	Open-Ended Question	100.00%	17

1	11/04/17 3:58PM ID: 55435956	Music sounded very harsh edged behind the main instrument
2	11/04/17 4:05PM ID: 55436275	it sounds nice and nothing seems off about it
3	11/04/17 4:32PM ID: 55438150	The added tones were out of sync
4	11/04/17 5:07PM ID: 55439931	The song was boring but nothing sounded 'wrong' with it
5	11/04/17 5:25PM ID: 55440542	The song crackled in parts
6	11/04/17 5:54PM ID: 55441998	felt more real

Why did you like/dislike this clip, and why did you feel this song was/was not technically correct?				
			Response Percent	Response Total
7	11/04/17 6:19PM ID: 55442822	Very soothing, I could meditate to this.		
8	11/04/17 8:24PM ID: 55447090	The song sounds too sad. But as a song it sounds well put together. There is a clicking sound when the chords change though.		
9	11/04/17 8:32PM ID: 55447809	Catchy tune		
10	11/04/17 8:39PM ID: 55448135	Depressing		
11	11/04/17 8:44PM ID: 55448146	Melodic with interesting top and bottom parts.		
12	11/04/17 8:55PM ID: 55448688	It was quite dull and boring. The sound of the music flowed very well.		
13	11/04/17 9:25PM ID: 55449746	This piece felt it was missing a smooth transition between different layers.		
14	12/04/17 12:46AM ID: 55454756	Don't like the two sounds together- wouldn't be sounds I'd use together. Sounds like it was technically correct, no noticeable mistakes.		
15	12/04/17 5:15AM ID: 55457960	Chordal overlay appeared to hesitate every few bars		
16	24/04/17 11:42AM ID: 56348925	I liked this song because it was chilled out and you can chill out to it after a hard days work. I thought the song was technically correct because nothing sounded out of place		
17	24/04/17 11:56AM ID: 56351120	The sound coming from the instrument is odd and dull sounding, it's lifeless		
			answered	17
			skipped	0

Sample 4 is played here. The sample can be played by following this URL: https://www.youtube.com/watch?v=SnTjg6gWzTc

Question 13

On a scale of 1 to 10, with 1 being strongly dislike and 10 being strongly enjoy, how much do you like this piece of music?			**Response Percent**	**Response Total**
1	1		0.00%	0
2	2		0.00%	0
3	3		0.00%	0
4	4		0.00%	0
5	5		23.53%	4
6	6		23.53%	4
7	7		17.65%	3
8	8		5.88%	1
9	9		29.41%	5
10	10		0.00%	0

Analysis								
	Mean:	6.94	Std. Deviation:	1.55	Satisfaction Rate:	66.01	answered	17
	Variance:	2.41	Std. Error:	0.38			skipped	0

Question 14

Did you think this song was technically correct? (with technically correct meaning that the sound of the song did not sound incorrect/as though an error had been made)				
			Response Percent	**Response Total**
1	Yes		88.24%	15
2	No		11.76%	2
Analysis	Mean: 1.12 · Std. Deviation: 0.32 · Satisfaction Rate: 11.76		answered	17
	Variance: 0.1 · Std. Error: 0.08		skipped	0

Question 15

Why did you like/dislike this clip, and why did you feel this song was/was not technically correct?			
		Response Percent	**Response Total**
1	Open-Ended Question	100.00%	17

1	11/04/17 3:58PM ID: 55435956	Background warbling is more in tune with the music however still distracts from the main piano piece
2	11/04/17 4:05PM ID: 55436275	sounds sad and calming, i'm not really sure what would make a clip correct or not
3	11/04/17 4:32PM ID: 55438150	The tones seemed to be in sync here
4	11/04/17 5:07PM ID: 55439931	Was a nice soothing song with no clashing sounds
5	11/04/17 5:25PM ID: 55440542	Yeah it seemed correct with a nice piano tempo
6	11/04/17 5:54PM ID: 55441998	getting better

Why did you like/dislike this clip, and why did you feel this song was/was not technically correct?				
			Response Percent	Response Total
7	11/04/17 6:19PM ID: 55442822	Always been fond of piano/synth.		
8	11/04/17 8:24PM ID: 55447090	Loved the song. Piano was great. Nice flute in the background. Sounds fine as the two instruments go together really well. Still a popping sound though.		
9	11/04/17 8:32PM ID: 55447809	I enjoyed the piece, soothing. I'm not sure about the flute sound it sounds sometimes like it doesn't fit.		
10	11/04/17 8:39PM ID: 55448135	Sounds like a love song		
11	11/04/17 8:44PM ID: 55448146	Do not think the harmonic technical sound melded well with the piano sound		
12	11/04/17 8:55PM ID: 55448688	It felt like it was a good melody that could be played with or without lyrics. Random notes disrupted the pattern.		
13	11/04/17 9:25PM ID: 55449746	Same as first two. Flowed nicely, maybe a layer was a tad overpowering against the gentler layer.		
14	12/04/17 12:46AM ID: 55454756	I liked the melody in this piece, sounded much better than the previous 3 versions. No noticeable mistakes in the song.		
15	12/04/17 5:15AM ID: 55457960	Think I'm becoming tired of listening to the slightly late/delayed chordal overlay (with its slow attack on each note). The chordal melody starting to sound contrived.		
16	24/04/17 11:42AM ID: 56348925	I like this song because I feel like I could go to sleep listening to this. I think this song is technically correct because the slow tempo of the melody matches the style of the backing track.		
17	24/04/17 11:56AM ID: 56351120	The flute sounded dull and mechanical		
			answered	17
			skipped	0

7.2. Appendix 2

Augmented Creativity Questionnaire 1 – Volunteers with a music knowledge of >= 5

Sample 1 is played here. The sample can be played by following this URL:
https://www.youtube.com/watch?v=MD8GLRpIF8U

Question 4

On a scale of 1 to 10, with 1 being strongly dislike and 10 being strongly enjoy, how much do you like this piece of music?

			Response Percent	Response Total
1	1		0.00%	0
2	2		0.00%	0
3	3		0.00%	0
4	4		20.00%	1
5	5		20.00%	1
6	6		0.00%	0
7	7		40.00%	2
8	8		20.00%	1
9	9		0.00%	0
10	10		0.00%	0
Analysis	Mean: 6.2	Std. Deviation: 1.47	Satisfaction Rate: 57.78	
	Variance: 2.16	Std. Error: 0.66		
			answered	5
			skipped	0

Question 5

Did you think this song was technically correct? (with technically correct meaning that the sound of the song did not sound incorrect/as though an error had been made)				
			Response Percent	**Response Total**
1	Yes		100.00%	5
2	No		0.00%	0
Analysis	Mean: 1 · Std. Deviation: 0 · Satisfaction Rate: 0		answered	5
	Variance: 0 · Std. Error: 0		skipped	0

Question 6

Why did you like/dislike this clip, and why did you feel this song was/was not technically correct?				
			Response Percent	**Response Total**
1	Open-Ended Question		100.00%	5
	1	11/04/17 5:25PM ID: 55440542	Sounded good to me	
	2	11/04/17 8:32PM ID: 55447809	It was an easy listening piece of music, calming. Sounds blended and flowed together.	
	3	11/04/17 8:44PM ID: 55448146	Movement through the guitar chord was distracting. The different instruments in the music seemed disjointed - no blend	
	4	12/04/17 5:15AM ID: 55457960	Lack of dischord & interesting left/right balance between synthesised / plucked string sounds.	
	5	24/04/17 11:42AM ID: 56348925	I liked this music because it felt like I was having a masarge and it felt relaxing. I thaught the clip was technically correct because it blended well together.	
			answered	5
			skipped	0

Sample 2 is played here. The sample can be played by following this URL: https://www.youtube.com/watch?v=DEXZGZ4XC2Q

Question 7

On a scale of 1 to 10, with 1 being strongly dislike and 10 being strongly enjoy, how much do you like this piece of music?				
			Response Percent	**Response Total**
1	1		0.00%	0
2	2		0.00%	0
3	3		0.00%	0
4	4		20.00%	1
5	5		40.00%	2
6	6		0.00%	0
7	7		20.00%	1
8	8		0.00%	0
9	9		20.00%	1
10	10		0.00%	0
Analysis	Mean: 6; Std. Deviation: 1.79; Satisfaction Rate: 55.56		answered	5
	Variance: 3.2; Std. Error: 0.8		skipped	0

Question 8

Did you think this song was technically correct? (with technically correct meaning that the sound of the song did not sound incorrect/as though an error had been made)

			Response Percent	Response Total
1	Yes		80.00%	4
2	No		20.00%	1
Analysis	Mean: 1.2, Std. Deviation: 0.4, Satisfaction Rate: 20		answered	5
	Variance: 0.16, Std. Error: 0.18		skipped	0

Question 9

Why did you like/dislike this clip, and why did you feel this song was/was not technically correct?

		Response Percent	Response Total
1	Open-Ended Question	100.00%	5

1	11/04/17 5:25PM ID: 55440542	The piece of music sounded upbeat but quite jumpy in parts.
2	11/04/17 8:32PM ID: 55447809	It was a lot more upbeat which isn't really my type of music. All sounds blended and sounded like they belonged in the music.
3	11/04/17 8:44PM ID: 55448146	More cohesive than clip 1. Seemed all technically produced and the different sounds blended well together
4	12/04/17 5:15AM ID: 55457960	Found chordal overlay tone to clash with stronger rhythym of more staccato synth
5	24/04/17 11:42AM ID: 56348925	I liked the music because the beat made me feel like I wanted to dance. I thought the music was technically correct because nothing sounded out of place

answered	5
skipped	0

Sample 3 is played here. The sample can be played by following this URL: https://www.youtube.com/watch?v=-9xN15aHQZk

Question 10

On a scale of 1 to 10, with 1 being strongly dislike and 10 being strongly enjoy, how much do you like this piece of music?				
			Response Percent	**Response Total**
1	1		0.00%	0
2	2		0.00%	0
3	3		0.00%	0
4	4		20.00%	1
5	5		0.00%	0
6	6		20.00%	1
7	7		60.00%	3
8	8		0.00%	0
9	9		0.00%	0
10	10		0.00%	0
Analysis	Mean: 6.2	Std. Deviation: 1.17	answered	5
	Variance: 1.36	Std. Error: 0.52	skipped	0
	Satisfaction Rate: 57.78			

Question 11

Did you think this song was technically correct? (with technically correct meaning that the sound of the song did not sound incorrect/as though an error had been made)				
			Response Percent	Response Total
1	Yes		100.00%	5
2	No		0.00%	0
Analysis	Mean: 1 · Std. Deviation: 0 · Satisfaction Rate: 0		answered	5
	Variance: 0 · Std. Error: 0		skipped	0

Question 12

Why did you like/dislike this clip, and why did you feel this song was/was not technically correct?				
			Response Percent	Response Total
1	Open-Ended Question		100.00%	5
1	11/04/17 5:25PM ID: 55440542	The song crackled in parts		
2	11/04/17 8:32PM ID: 55447809	Catchy tune		
3	11/04/17 8:44PM ID: 55448146	Melodic with interesting top and bottom parts.		
4	12/04/17 5:15AM ID: 55457960	Chordal overlay appeared to hesitate every few bars		
5	24/04/17 11:42AM ID: 56348925	I liked this song because it was chilled out and you can chill out to it after a hard days work. I thought the song was technically correct because nothing sounded out of place		
			answered	5
			skipped	0

Sample 4 is played here. The sample can be played by following this URL: https://www.youtube.com/watch?v=SnTjg6gWzTc

Question 13

On a scale of 1 to 10, with 1 being strongly dislike and 10 being strongly enjoy, how much do you like this piece of music?

			Response Percent	Response Total
1	1		0.00%	0
2	2		0.00%	0
3	3		0.00%	0
4	4		0.00%	0
5	5		20.00%	1
6	6		20.00%	1
7	7		20.00%	1
8	8		0.00%	0
9	9		40.00%	2
10	10		0.00%	0

Analysis	Mean:	7.2	Std. Deviation:	1.6	Satisfaction Rate:	68.89	answered	5
	Variance:	2.56	Std. Error:	0.72			skipped	0

Question 14

Did you think this song was technically correct? (with technically correct meaning that the sound of the song did not sound incorrect/as though an error had been made)				
			Response Percent	**Response Total**
1	Yes		100.00%	5
2	No		0.00%	0
Analysis	Mean: 1; Std. Deviation: 0; Satisfaction Rate: 0		answered	5
	Variance: 0; Std. Error: 0		skipped	0

Question 15

Why did you like/dislike this clip, and why did you feel this song was/was not technically correct?				
			Response Percent	**Response Total**
1	Open-Ended Question		100.00%	5
1	11/04/17 5:25PM ID: 55440542	Yeah it seemed correct with a nice piano tempo		
2	11/04/17 8:32PM ID: 55447809	I enjoyed the piece, soothing. I'm not sure about the flute sound it sounds sometimes like it doesn't fit.		
3	11/04/17 8:44PM ID: 55448146	Do not think the harmonic technical sound melded well with the piano sound		
4	12/04/17 5:15AM ID: 55457960	Think I'm becoming tired of listening to the slightly late/delayed chordal overlay (with its slow attack on each note). The chordal melody starting to sound contrived.		
5	24/04/17 11:42AM ID: 56348925	I like this song because I feel like I could go to sleep listening to this. I think this song is technically correct because the slow tempo of the melody matches the style of the backing track.		
			answered	5
			skipped	0

7.3. Appendix 3

Augmented Creativity Questionnaire 2

Sample 1 is played here. The sample can be played by following this URL: https://www.youtube.com/watch?v=MD8GLRpIF8U

Question 1

1. Consider the flute sounding instrument in the above video: Do you think this melody sounds out of sync with the rest of the audio?

			Response Percent	Response Total
1	Yes		23.53%	4
2	No		76.47%	13

Analysis							
Mean:	1.76	Std. Deviation:	0.42	Satisfaction Rate:	76.47	answered	17
Variance:	0.18	Std. Error:	0.1			skipped	0

Question 2

2. Consider the flute sounding instrument in the above video: Do you think the instrument played any notes that clashed with the existing music?

			Response Percent	Response Total
1	Yes		17.65%	3
2	No		82.35%	14

Analysis							
Mean:	1.82	Std. Deviation:	0.38	Satisfaction Rate:	82.35	answered	17
Variance:	0.15	Std. Error:	0.09			skipped	0

Question 3

3. Consider the flute sounding instrument in the above video: Did you like the overall sound of the instrument?				
			Response Percent	**Response Total**
1	Yes		58.82%	10
2	No		41.18%	7

Analysis	Mean:	1.41	Std. Deviation:	0.49	Satisfaction Rate:	41.18	answered	17
	Variance:	0.24	Std. Error:	0.12			skipped	0

3. Listen to Audio Sample 2

Question 4

4. Consider the flute sounding instrument in the above video: Do you think this melody sounds out of sync with the rest of the audio?				
			Response Percent	**Response Total**
1	Yes		11.76%	2
2	No		88.24%	15

Analysis	Mean:	1.88	Std. Deviation:	0.32	Satisfaction Rate:	88.24	answered	17
	Variance:	0.1	Std. Error:	0.08			skipped	0

Question 5

5. Consider the flute sounding instrument in the above video: Do you think the instrument played any notes that clashed with the existing music?

			Response Percent	Response Total
1	Yes		11.76%	2
2	No		88.24%	15

Analysis	Mean:	1.88	Std. Deviation:	0.32	Satisfaction Rate:	88.24	answered	17
	Variance:	0.1	Std. Error:	0.08			skipped	0

Question 6

6. Consider the flute sounding instrument in the above video: Did you like the overall sound of the instrument?

			Response Percent	Response Total
1	Yes		52.94%	9
2	No		47.06%	8

Analysis	Mean:	1.47	Std. Deviation:	0.5	Satisfaction Rate:	47.06	answered	17
	Variance:	0.25	Std. Error:	0.12			skipped	0