AR-Live Score:
Live Musical Scores in Head-Mounted
Augmented Reality

Zeruo Liu

A report submitted as part of the assessment for
An Independent Studies 12 units (COMP8755) in
The Research School of Computer Science
Australian National University

2/June/2020
Except where otherwise indicated, this report is my own original work.

Zeruo Liu

2/June/2020
Acknowledgment

Working on this project took 2 semesters. I want to show my gratitude to the people who have helped me and contributed to the project.

Firstly, I wish my best appreciation to my ANU supervisor, Henry Gardner, and Charles Martin, who provided me with constant academic supports for this project. I would like to thank them for helping me get into the music field and show me how to think about innovative ideas in a project. Henry led me to the research of HCI, AR, and music field. Charles instructed me much in music retrieval technology and HCI research design ideas.

I would also like to give my thanks to my CSIRO line manager and supervisor, Stuart Anderson, and Mingze Xi. They take a large amount of time to instruct me on the technology of Augmented Reality in HoloLens, version control, and report writing skills. Thanks for them always be patient and supportive when I ask questions and provide me opportunities to be trained in CSIRO immersive lab. They support me in the COVID-19 situation and lent me all the devices I need.

Finally, I want to say thanks to my parents and my grandpa who give me encouragement every week. Also, my friend Xuanying and boyfriend Yuchen are always by my side, give me some thoughts, and be my listeners through this year.
Abstract

This project describes an application that displays real-time musical scores and uses intelligent music recognition in an Augmented Reality (AR) environment to provide a better experience for musicians during performance or composition. Compared with using sheet music on a music stand, the advantage of the combination of head-mounted AR and intelligence is to free the hands of a musician in performance, to free up space that would otherwise be used by music stands, to provide a better 3D visualization experience and to providing convenience for musicians. An early prototype, known as AR-LiveScore, has been developed for a Microsoft AR device. This prototype is made up of three real-time functions: (i) 3D abstract visualization (“GraphMusic”), (ii) a music file processing, intelligent music construction and music tracking system (“PlayScore”) and (iii) pitch recording (“WriteScore”). The interactions and scenes are set in the AR headset, while the spectral flux and pitch detection technology are implemented for real-time audio analysis. A self-evaluation has been used to test the sensitivity and completeness of the application, and a user study has been conducted to test the perceived usefulness of this application by musicians. The ideas of having a 3D musical score and intelligence in an AR music application are both novel in the literature and potentially useful for musicians.
# Contents

## Table of Contents

Acknowledgment ........................................................................................................ iii

Abstract ......................................................................................................................... iv

Contents ......................................................................................................................... v

List of Figures ................................................................................................................ ix

1. **Introduction** ............................................................................................................. 13
   1.1 Introduction to Project ............................................................................................. 13
   1.2 Motivation ............................................................................................................... 14
   1.3 Structure of Report ................................................................................................. 15

2. **Background** ........................................................................................................... 16
   2.1 Augmented Reality ................................................................................................. 16
   2.2 Microsoft HoloLens ............................................................................................... 17
   2.3 Unity ....................................................................................................................... 17
   2.4 Mixed-Reality Toolkit ............................................................................................ 18
   2.5 Muse Score ............................................................................................................. 18
   2.6 GarageBand ............................................................................................................ 18
   2.7 Music Performance ............................................................................................... 19
   2.8 Music Representation in Augmented Reality ....................................................... 20
Contents

2.8.1 Music Notation for Performance ................................................................. 20
2.8.2 Graph Music for AR Visualization ............................................................ 22
2.8.3 Innovative Music Representation ............................................................... 23
2.9 Intelligent System in Music ........................................................................... 24
2.10 Music Recognition ....................................................................................... 25
  2.10.1 Onset Detection ..................................................................................... 26
  2.10.2 Frequency Identification ....................................................................... 26
  2.10.3 Improvisation ....................................................................................... 26
3 Development Environment .............................................................................. 29
  3.1 Software ....................................................................................................... 29
  3.2 Hardware ..................................................................................................... 30
  3.3 Scene Requirements .................................................................................... 31
  3.4 Working Environment ................................................................................ 31
4 Implementation ............................................................................................... 34
  4.1 MusicXML File Process ............................................................................... 34
  4.2 Live 3-D Music Scores ............................................................................... 36
  4.3 Music Detection for Pitch and Rhythm ...................................................... 38
    4.3.1 Music Visualizer (GraphMusic) ............................................................ 39
    4.3.2 Onset Detection ................................................................................... 41
    4.3.3 Frequency Detection ............................................................................ 42
## Contents

4.4 AR Interactions ........................................................................................................45

4.5 Visual Feedback .........................................................................................................47

4.6 Summary of Implementation .....................................................................................48

5 Limitations ....................................................................................................................52

5.1 Music Sheet Representation .......................................................................................52

5.2 Music Detection ..........................................................................................................52

5.3 Interactions ................................................................................................................53

6 Evaluation .....................................................................................................................55

6.1 Self-evaluation ............................................................................................................55

6.1.1 Music Representation Evaluation ........................................................................55

6.1.2 Music Recognition Evaluation ..............................................................................59

6.2 User study ..................................................................................................................60

6.2.1 Basic Information ..................................................................................................61

6.2.2 Music Visualizer ...................................................................................................62

6.2.3 Music representation and tracking .......................................................................62

6.2.4 Music composition ...............................................................................................63

6.3 Summary of Evaluation ............................................................................................64

7 Conclusion and future work .....................................................................................67

7.1 Conclusion ................................................................................................................67

7.2 Future work ................................................................................................................68
Bibliography ............................................................................................................................................. 69

Appendix A Milestone ............................................................................................................................. 74

Appendix B Individual Study Contract ................................................................................................. 75

Appendix C Artefacts ............................................................................................................................. 78

Appendix D Improvisation Prototypes ................................................................................................. 80
  • Music Synthesizer ................................................................................................................................. 80
  • Sound Analysis ..................................................................................................................................... 81
  • Harmonious Representation .................................................................................................................. 82

Appendix E Information Sheet ............................................................................................................... 84

Appendix F Consent form ....................................................................................................................... 88

Appendix G Questionnaire .................................................................................................................... 89
List of Figures

Figure 1 A simplified representation of Reality Virtual Continuum ........................................ 16
Figure 2 The real application of hand-free HoloLens (Microsoft, 2019) ................................. 17
Figure 3 GarageBand operation layout screenshot on mac................................................... 19
Figure 4 3D notation representation structure in HoloLens (David & Carey, 2019) .......... 20
Figure 5 2D music sheet in HoloLens with a bouncing ball as music pointer ................. 21
Figure 6 Band accompany demonstration in “Music Everywhere” for piano (HoloLens app) ......................................................................................................................................................................................... 22
Figure 7 AR Music Visualization (Labs, 2017) (Shantanu, et al., 2017) ....................... 22
Figure 8 Another smooth AR Music visualization (Carter, 2016) ...................................... 23
Figure 9 A circular music visualizer layout with visualizers (Macedo, 2017) .............. 23
Figure 10 Gestures to control AR Groove ............................................................................. 24
Figure 11 A diagram to show normal music intelligent tools (Hirokazu & Mark, 2019). 25
Figure 12 Improvisation showing harmonious notes in HoloLens (Shantanu, et al., 2017) ......................................................................................................................................................................................... 27
Figure 13 The Alienware Laptop I use to develop the software and video the demonstration ......................................................................................................................................................................................... 32
Figure 14 The Microsoft HoloLens2 I use to video and develop AR application .......... 32
Figure 15 The electric piano I use to test the AR app and it is also used to video in the demo as the real musical instrument ......................................................................................................................................................................................... 33
List of Figures

Figure 16 Layout in Muse Score to edit a music sheet before saving it to MusicXML and then importing into the AR music visualization system of this report. ..................35

Figure 17 Layout of GarageBand used for editing a midi file to test AR-LiveScore .......36

Figure 18 The 3D sheet music I create from MusicXML file automatically ..................37

Figure 19 The 3D sheet music I create from a MusicXML file automatically once it is read into PlayScore .................................................................37

Figure 20 The layout of 3D symbols after drawing music sheet in the Unity editor play mode.........................................................................................38

Figure 21 Workflow of pitch detection..................................................................39

Figure 22 Music visualizer as frequency bands ..................................................41

Figure 23 Main menu for selecting the sub-scene in play mode .......................46

Figure 24 Main menu for selecting the “Practice” scene and show music sheet choices..46

Figure 25 The layout of real-time pitch recording with small circle on five-line sheet. ...48

Figure 26 The first function is to visualizes music in frequency bands in real time. ....49

Figure 27 The second function displays a 3D music sheet from a MusicXML file. An audio source, such as a performing musician, can then be tracked in real time and their place in the music can be displayed. .................................................................50

Figure 28 The third function records a pitch being played in real time and visualizes this as music on the 3D. .................................................................51

Figure 29 A diagram to show the process and results of the user study evaluation ........65

Figure 30 the screenshot of main scene with the coding structure on the left, scene settings at the bottom and Hierarchy structure of this scene on the right. .........................78
List of Figures

Figure 31 the screenshot of “Write Score” scene with the coding structure on the left, scripts listed at the bottom and Hierarchy structure of this scene on the right. .....................79

Figure 32 the screenshot of “Play Score” scene with the coding structure on the left, scenes listed at the bottom and Hierarchy structure of this scene on the right. .....................79

Figure 33 Music synthesizer in 3D version ......................................................................81

Figure 34 an example of showing 3 highest notes at one time together for group musicians’ improvisation ..............................................................................................................82

Figure 35 The sheet music shows the main rhythm in black notes while red notes are hint notes ..............................................................................................................................................82

Figure 36 A piano format of showing the main rhythm in read and harmony suggestion in blue ........................................................................................................................................83

Figure 37 A string format of showing the hints in other points ........................................83
Table 1 Software used in this project ................................................................. 29

Table 2 Hardware apparatus used in this project .................................................. 30

Table 3 Note and frequency setting comparison chart for the frequency detection in Octave 4 (from C4-B4) (mwguitars, 2013) ................................................................. 43

Table 4 Note and frequency setting comparison chart for the frequency detection in Octave 5 (from C5-B5) (mwguitars, 2013) ................................................................. 44

Table 5 Evaluation of sheet music elements as a sheet system (Dolmetsch, 2019) ........ 55

Table 6 Detailed music notes evaluation (Dolmetsch, 2019) ................................. 57

Table 7 Detailed rest note evaluation (Dolmetsch, 2019) ....................................... 58

Table 8 The evaluation of music recognition algorithm (120bpm is set as the maximum test tempo for the self-evaluation since 120bpm is relatively fast tempo). ................. 60
1. Introduction

1.1 Introduction to Project

This project explores live musical notation in a head-mounted Augmented Reality (AR) environment. The overall vision of this work is to develop intelligent tools for the display of live and visual feedback (including musical score notation) to musicians during music performance, rehearsal, and composition. A system that utilizes tools such as those discussed in this report and developed in this project would need to have three main requirements.

- The system needs to be able to “listen to music” as it is being played. This means that real-time signal analysis needs to be combined with tools that convert an audio stream into real-time music – where music is understood as being a collection of music data including musical tones and rhythms and so on.
- The system needs to be able to “understand music” as it is being performed. This means that real-time music with “understanding” of real-time audio data, musical scores, and musical file formats. Such understanding could involve aspects of audio data analysis and artificial intelligence. Also, the “understanding” could include automatic file processes.
- The system needs to be able to represent “live” visual feedback to performing musicians in useful ways.

AR headsets afford new possibilities of use for performing musicians in that they can be worn in a hands-free manner. Thus, musicians are able to play a musical instrument while looking through an augmented display of the real world. Such an augmented display could represent information to provide real-time feedback to the performing musicians.
This project develops ideas that were articulated in an earlier project and paper (Zeruo, et al., 2019). In that earlier work, a hologram of musical notation (an image score of conventional music) was displayed in an AR headset. This 2D music hologram was able to be positioned in space in arbitrary ways relative to the musician and a demonstration was shown of some real-time visual feedback – that of a bouncing ball pointer moving along the musical score according to the in-built timer. It is important to note that, in that earlier work, the real-time visual feedback was “canned” rather than being “live”. That is, the real-time movement of the pointer was faked to move in accordance with an internal timer rather than being responsive to actual music as it was being played. In the earlier work, feedback was obtained from a focus group of musicians. An important aspect of that the feedback was the interest of those musicians in seeing whether such a system would be able to usefully provide live musical feedback that was genuinely responsive to music being played (Zeruo, et al., 2019).

1.2 Motivation

There has been work about visualizing 2D and 3D music sheets in head-mounted device which will be explained in detail in Chapter 2.2. AR can provide new possibilities to enhance music performance and playing experience for musicians. With AR devices, it is possible for players to read the music sheet without turning over the pages and interacting with other cooperative players or conductors (Zeruo, et al., 2019) while playing music or improvise during the music performance.

However, most of the audio-related work in the AR device is implemented without synchronizing the real-time music and music sheet and looks fake and mechanical (evltube, 2020). Synchronizing the music with music notes displayed will lead the users to play more comfortably in the AR environment and give a more flexible performance for audiences. This system will potentially allow the user to perform in their own mood according to the musician’s comprehension of music and they can control the playing speed on the stage.
Musicians also need an intelligent system to update their own music sheet as a practically used system which is potentially useful. This process can be separated as importing the music file and constructing music sheets automatically visualization in the headset. Thus, we develop a smart music system as an early prototype to make visualization of the music sheet in AR headset more intelligent and automatic in this project.

### 1.3 Structure of Report

Chapter 2 reviews the background of the project, literature review, and the related technology in this project.

Chapter 3 presents the required development environment including several software and hardware settings for this project.

Chapter 4 listed the detailed implementations and designs for the AR application.

Chapter 5 explains several technical problems which have been met and present the limitations of the AR application.

Chapter 6 presents a detailed self-evaluation and a user study in this project and list out the user study feedback.

Chapter 7 shows the conclusion of this project and list the potential future work.

In the end, the appendix of milestones in this project, the real screenshots, my prototype video link, improvisation paper prototype, and evaluation protocol are listed as a demonstration.
2 Background

2.1 Augmented Reality

As shown in Figure 1, the Reality Virtual Continuum (Milgram et al., 1995) shows extreme of the left which only shows the real scene, while the right extreme is the sole Virtual Reality field. AR is Mixed Reality between two extremes. In detail of mixed reality, Augmented Reality is the technology adding virtual graphics or elements to reality which is much closer to the reality side. Augmented Reality environments are original real but with virtuality, in the sense that they have been created artificially, by computer (Milgram et al., 1995), while Augmented Virtuality is adding “Reality” to virtual mapping environment.

![Mixed Reality Continuum](image)

**Figure 1 A simplified representation of Reality Virtual Continuum**

(Paul, et al., 1995)

The concept of augmented reality was initially proposed decades ago, but has only started entering our life recently with accessible technologies being released, such as Microsoft Mixed Reality Toolkit, Google AR Core, Apple ARKit, etc. AR has the potential to vastly improve the quality of daily life in different fields.
2.2 Microsoft HoloLens

HoloLens is an untethered AR device combining applications and solutions together across all fields (Microsoft, 2019). It combines sophisticated hardware design to provide a holographic experience for users. The HoloLens 2 is already well developed with the helpful Mixed-Reality Toolkit (Details in Section 2.4).

In comparison with the original HoloLens, more functions such as eye gaze tracking, two-handed fully articulated model tracking, holographic photos and videos have been developed for HoloLens 2. The HoloLens 2 is the AR headset that has been used in this project as Figure 2 shown.

2.3 Unity

Unity is a tool engine to create 2-dimensional and 3-dimensional games and it has implementations for many companies and operating systems in phones, computers, and game (Unity - Manual: Unity User Manual (2018.2)).
Unity supports development for many AR platforms, including HoloLens, Meta, iOS and Android devices with different frameworks and Software Development Kits (SDK). Therefore, Unity is chosen to develop the HoloLens AR application.

2.4 Mixed-Reality Toolkit

The Mixed-Reality Toolkit (MRTK) is an open source code library driven by Microsoft. It supports several Windows Mixed-Reality platforms including HoloLens, making it perfectly fit for this project. The listed features are Input System, Voice Commanding, Gaze, Select, Visualization, UI Controls, Solver and Interactions, Spatial Understanding and Diagnostic Tool. Voice commands, gestures, cursor, and selections are implemented in the source code. This library is configured in Unity for my AR application.

2.5 Muse Score

Muse Score is a music score writer and a shared platform which can be edited on most of the devices (MuseScore, 2020). The main purpose of this software is to produce unlimited musical scores for users. It supports a variety of functions including music sheet edit, MIDI input and output, tablature, percussion notations and nearly every implementation in music sheets. Also, it is supported different file formats such as MusicXML which is the music file source format used in this project.

2.6 GarageBand

GarageBand is a digital music creation program provided by Apple (Apple, 2020). There are 2 functions that are MIDI editing and instrument synthesizer applied in this project. It has pre-set for pianos, strings, guitar and voice, and an incredible selection of session drummers and percussionists. The editing can be conducted transposition both audio and MIDI smoothly including pitch, velocity, and duration. All these audios are interactive tracks as Figure 3 shown.
2.7 Music Performance

Music performance requires strong internal psychology and external convenient environment for music players. We want AR applications to satisfy the players’ psychological requirements and to provide helpful AR scenes – especially for performance. There are different factors in terms of musicians’ moods and motions for musicians’ requirements for music performance (JUSLIN, 2003). Musicians’ moods and individual control of the music can be varied among musicians’ music performance, so an innovative presentation of music in performing music might enhance the performance quality. In a performance scenario, players should not be restricted by space size and omissible motions like turning over the music sheet. AR headsets can avoid these motions and provide users visualization of a clear music sheet. An intelligent system in AR headset can help with the clarification of a clear musical structure and make it easier for musicians to convey emotions by stylistic performance such as timing and emotions.
2.8 Music Representation in Augmented Reality

2.8.1 Music Notation for Performance

There have been some conventional and novel ideas in AR for music performance in the literature and on the internet. Some novel designs for music representation are designed as a new music representation. A MAX (a visual programming tool for music and multimedia) (MAX, 2020) developed 3D model is built to provide more possibilities in AR music (David & Carey, 2019) as Figure 4 shown. To visualize in HoloLens, Max communicates with the Unity3D platform. They designed the structure that with different node colour, line length and curvature represent articulative, temporal, and timbre properties separately. The various pitches are denoted by coloured lines. Performers explore various pathways through the score as dots are transformed during the performance (David & Carey, 2019).

Figure 4 3D notation representation structure in HoloLens (David & Carey, 2019)
Background

Most musicians still use traditional sheet music forms and instruments for performance. For instance, the familiar five-line staff in classical music. The five-line staff is visually composed of dots and lines (Andrea & Gordon, 1982). The music notation in this project is a traditional staff which is a set with five horizon lines and four spaces (Wikipedia, 2020) span in 3d space. Musicians can convert the music notations with “lines” and “dots” into meaningful and unambiguous information about the piece of music in terms of rhythm, phrasing, loudness, and expression. Therefore, it is important to represent suitable music notations for musicians during their performance. Music implemented in this HoloLens application is a conventional five-line staff. There has been evidence that there have been digital music stands marketed to replace the printed paper, and many musicians are becoming more comfortable with the digital display (Jason, 2008). Developers have abilities to create dynamic musical scores containing those music notes with powerful graphics using an intuitive development environment. Thus, we will implement digitally in AR with conventional five-line staff.

There are several demonstrations to show 2D five-line staff music sheet in 3-D space with HoloLens (Zeruo, et al., 2019) (evltube, 2020). These early prototypes with 2D images in headset prove the potential to explore more possibilities to optimise the reading music function in the headset. It is mentioned that the high synchronization and an intelligent system are required to substitute the fake pointer with a higher-quality user interactive experience.
2.8.2 Graph Music for AR Visualization

3D music visualization process employing a novel method of real-time reconfigurable control of 3D geometry and texture, employing blended control combinations of software oscillators, computer keyboard and mouse, audio spectrum, control recordings and MIDI protocol (Srini, et al., 2006). In this section, we give some AR applications for graph music visualization.

Figure 6 Band accompany demonstration in “Music Everywhere” for piano (HoloLens app)

Figure 7 AR Music Visualization (Labs, 2017) (Shantanu, et al., 2017)
As Figure 7 and Figure 8 shown, we can see there has been work in music visualization rendered with colours and synchronized with the playing of build-in music. Build-in music can be visualized by audio data and represent in different 3D object representation and colours. This is shown in 3D space first and we can add colour, transparency, and intensity as 6D world as Figure 8 is shown (Carter, 2016) which is more suitable for 3D AR and VR world than the 2D screen to represent. In this project, a visualization is designed to help users understand the audio source in gathered in real time while they are playing as visual feedback.

Figure 8 Another smooth AR Music visualization (Carter, 2016)

Figure 9 A circular music visualizer layout with visualizers (Macedo, 2017)

2.8.3 Innovative Music Representation

There is another innovative virtual music creator named AR Groove (Ivan, et al., 2001) as Figure 10 shown. Manipulations (Move and rotate) of AR Groove allows users to create
their own electronic music and interactively remix music for a music performance. This paper supports the ideas of novel music performance in “computer-supported improvisation” in the AR environment instead of precise and repeatable creation.

Figure 10 Gestures to control AR Groove

2.9 Intelligent System in Music

The intelligent system in the music industry is has been applied in music production to help composition, music performance, and improvisation but there are few of them in the AR and VR field. There has been researched on intelligent system on audio processing (Markus, et al., 2016) which is similar to the first step of the design of my application. An intelligent music system includes audio input, analysis, and output as Figure 11 shown. An intelligent system contains an engine to observe the environment such as “real-time audio input” or
“users” interactions” to make decisions about what visual feedback to give for the intelligent agents (Hirokazu & Mark, 2019).

With appropriate analysis and humans’ assessments, AR can be explored in developing intelligent music system since no related research has been done in this area.

2.10 Music Recognition

Music recognition has a lot of variations in terms of deep learning or simplified algorithm. It can also be real-time or search the terminated music file in the library. We use real-time analysis for the audio in this project.

Several pitch detection algorithms are implemented in this application for interactive computer music performance. Fundamental frequency detection is a well-researched problem in the case of recorded monophonic voices and instruments. However, the identification of real-time pitch becomes much more difficult in musicians’ performances like live concert settings (Cuadra, et al., 2001).

There are many possibilities to observe that errors occur in real-time music tracking. The need to avoid initial pitch-tracking errors in interactive music applications generates the following list of requirements for real-time algorithms (Cuadra, et al., 2001):

1. Ability to function in real-time
2. Minimal output delay (latency)
3. Accuracy in the presence of noise
Background

4. Sensitivity to musical requirements of the performance.

My music recognition algorithm is mainly implemented using onset detection and frequency detection for each musical note.

2.10.1 Onset Detection

“Onset” describes when a note starts. It happens when playing a note, which is triggered until it reaches its maximum amplitude followed by a decay phase. Various methods have been proposed for detecting the onset times of musical notes in audio signals using spectral features such as magnitude, phase, and complex domain representations (Dixon, 2006). The main method used in my application is Spectral Flux. Spectral flux measures the change in magnitude in each audio frequency bins. The big change of the magnitude represents that onset has been triggered.

2.10.2 Frequency Identification

Frequency identification in this project is to match the pitch for music recognition as a pitch detector. This application has been designed to detect the monophonic frequency of real-time audio sources in Unity. The underlying method is to detect the peak of the audio frequency spectrum data. The peak detection is basically to compare the last peak and the current one and set a threshold to multiply sample time. The result is the time between the two peaks, and the frequency is the reciprocal of the time (LeCroy, 2020).

2.10.3 Improvisation

Music improvisation is that a group of music players or an individual musician composes new music and spontaneously plays in real-time without specific rules (Hoon Hong, 2019). Improvisation is a multi-dimensional activity requiring extemporaneous creative music performance (Michele, 2017). We can see there are few applications and related thoughts about helping musicians to improvise in Mixed Reality environments. As Figure 12 shows, the “Music Everywhere” application, mentioned in 2.8, has a related design in show harmonising notes overlaying the piano keys and these harmonising notes have been chosen by the intelligent system according to the background chords (Shantanu, et al., 2017).
Figure 12 Improvisation showing harmonious notes in HoloLens (Shantanu, et al., 2017)
3 Development Environment

3.1 Software

Software used in this project are listed in Table 1 with their names, versions and detailed descriptions.

Table 1 Software used in this project

<table>
<thead>
<tr>
<th>Software Item</th>
<th>Version</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows SDK</td>
<td>Windows 10</td>
<td>To develop apps for Windows Mixed Reality headsets, Windows 10 Fall Creators Update is required.</td>
</tr>
<tr>
<td>Unity</td>
<td>2019.3.7f1</td>
<td>The Unity 3D engine provides supports for building mixed reality projects in Windows 10.</td>
</tr>
<tr>
<td>Visual Studio</td>
<td>2019</td>
<td>Visual Studio is used for code editing, deploying and building HoloLens Universal Windows Platform (UWP) application packages.</td>
</tr>
</tbody>
</table>
Development Environment

<table>
<thead>
<tr>
<th>Mixed Reality Toolkit</th>
<th>Mixed Reality Toolkit v2.3.0</th>
<th>A relatively stable Mixed-Reality HoloToolkit is implemented on HoloLens suitably.</th>
</tr>
</thead>
<tbody>
<tr>
<td>GarageBand</td>
<td>GarageBand (Mac)</td>
<td>The music creation studio in Mac to edit and synthesize music</td>
</tr>
<tr>
<td>Muse Score</td>
<td>Muse Score 3 (Mac)</td>
<td>The music application inside Mac to edit music to produce MusicXML and MIDI files</td>
</tr>
</tbody>
</table>

3.2 Hardware

Hardware devices used in this project are listed in the Table 2 with their names and descriptions.

Table 2 Hardware apparatus used in this project

<table>
<thead>
<tr>
<th>Hardware Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HoloLens 2 (Commercial)</td>
<td>HoloLens 2 is used in this project</td>
</tr>
<tr>
<td>Alienware laptop</td>
<td>Alienware Area-51 laptop is used to design the application</td>
</tr>
</tbody>
</table>
Development Environment

| Piano | Roland Electric Piano FP-90 for testing the music recognition performance |

3.3 Scene Requirements

The application requires a music playing scene without noises for HoloLens and machine to obtain and analyse real-time sound smoothly. The scene should also have enough space to hold the music sheets to visualize, since the buttons can be triggered only when the virtual button is not occluded by the real objects for detecting the gestures.

3.4 Working Environment

In the first semester, my working environment is in CSIRO immersive lab. In the second semester, I worked at home on account of the COVID-19 pandemic. CSIRO lend me all the equipment to support me work at home.

My working environment is equipped with an Alienware laptop (with software requirements listed in 3.1), a Microsoft HoloLens 2, and an electrical piano. The environment of this project is shown as these figures, including Figure 13 The Alienware Laptop I use to develop the software and video the demonstration, Figure 14, and Figure 15 The electric piano I use to test the AR app and it is also used to video in the demo as the real musical instrument. and these are photos at my home.
Figure 13 The Alienware Laptop I use to develop the software and video the demonstration

Figure 14 The Microsoft HoloLens2 I use to video and develop AR application
Figure 15 The electric piano I use to test the AR app and it is also used to video in the demo as the real musical instrument.
4 Implementation

My intelligent-AR software implementation called AR-Live Score is aimed at helping users to visualise the music sheets and to get visual interactions from the automatic analysis of audio sources in real time.

This section discusses the detailed implementation and describes how the software has been designed and developed in Unity, Music studio and the HoloLens MRTK.

My AR-Live Score application is 3 separate functions and they are (i) PlayScore (Music file parsing, 3D sheet drawing, and music tracking), (ii) WriteScore (Pitch Recording) and (iii) GraphMusic (Abstract visualization in graphics).

4.1 MusicXML File Process

MusicXML is a music file format for music experts to edit music scores (makemusic, 2020). It is an XML-based file format, and we can edit a MusicXML file using software such as MuseScore (MuseScore, 2020). In the present prototype, we currently assume that the pitch range of the music is limited to be in octaves 4 and 5 and we have constructed MusicXML files to test the prototype which is limited to this range.

The prototype reads and loads MusicXML files using streaming asset folder loading function in Unity3D (Unity, 2020). The tempo of a piece of music is the speed of the underlying beat, so we need to set the “tempo map” first and then place the symbols according to the tempo map. A tempo map is gathered from beats per minute (bpm) from MusicXML file. After that, we start to arrange all the elements on the 3D music sheet.
Implementation

According to this tempo map. For each element (such as a division, beat, chord, title, work time, clef, sign, step, type, duration), the software has a switch statement that considers all the cases that if it is a symbol, a bar or a note and then we can define the duration and type of the symbol or note. Symbols, bars, and notes are stored one by one in several separate dictionary structures. We can get access to the stored dictionaries from this file for the next process.

![Figure 16 Layout in Muse Score to edit a music sheet before saving it to MusicXML and then importing into the AR music visualization system of this report.](image)

1 There is a related resource on how to parse MusicXML file on GitHub (Scripts/parseScripts/xmlParser/XmlParser.cs) as a reference (qy, 2017)
Implementation

Figure 17 Layout of GarageBand used for editing a midi file to test AR-LiveScore

4.2 Live 3-D Music Scores

3-D models of music scores and all of the music symbols prefabs have been used in this project to build a complete 3-D environment experience for users. The models implemented mainly are 3-D notes for different time-length, music sheet lines, bar lines, and rest symbols for different duration, treble, and bass clef which are all deactivated default in the scene.

When AR-LiveScore scene starts, before starting to construct a scene, a musical staff made up of five white lines is drawn in front of the camera view. After reading information about each music component from the MusicXML file, a symbol corresponding to this particular component is activated and shown in its position on the staff.

The position of the musical symbols is horizontally arranged from left to right. After adding a symbol, the horizontal position is calculated by adding 0.2 meters locally (the localPosition variable in Unity (Unity, 2020)). Treble and bass clef symbols, bars, rest symbols, and the tracking pointer are also displayed using a similar vertical spacing by default. The vertical position of notes is decided by the pitch (an octave and step data) of each note.

All the drawing is started immediately when users open the PlayScore application.
Figure 18 The 3D sheet music I create from MusicXML file automatically

Figure 19 The 3D sheet music I create from a MusicXML file automatically once it is read into PlayScore
4.3 Music Detection for Pitch and Rhythm

Music detection in this section is to help to control the movement of music pointer for the tracking and display the real-time playing notes in the PlayScore application. For music recognition, we choose to get real-time data to match the data gathered from MusicXML file mentioned before. The diagram demonstrates how to recognize music in real-time as Figure 21 shown. We first get the real-time audio input data and do some transformation for the following steps. After that, the onset detection is tested. If there is an onset, we can calculate the frequency and check if the frequency is matched the information of the next note. If it is matched, we can continue to analyse the data and repeat every algorithm for the next note.
4.3.1 Music Visualizer (GraphMusic)

This function detects real-time audio and gives some feedback to the users. Before checking the onset and frequency, the software needs to detect real-time audio input through the microphone and make it a temporary audio track. Microphones are checked to get access if detected. If there is no microphone, the real-time detection cannot be conducted. If there is more than one microphone, the software will choose the first microphone detected for the audio analysis. Then the track is collected and transformed to audio frequency formats in using the “get spectrum data” Unity function which performs a fast Fourier transform (Unity, 2020):

```csharp
public void AudioSource.GetSpectrumData(float[] samples,
                                        int channel, FFTWindow window)
```
In GraphMusic, the sample rate is set to be 44100 which is the minimum sample rate in Unity. We set the FFT window to transform from the original wave domain to the representation of frequency domain when the in-built sequence length is an integer power-of-two. Blackman-Harris window is set as:

\[
W[n] = 0.35875 - (0.48829 \times \cos(1.0 \times n/N)) + (0.14128 \times \cos(2.0 \times n/N)) - (0.01168 \times \cos(3.0 \times n/N))
\]

N in this formula is the window length and this function returns an N-point symmetric Blackman-Harris window \((W[n])\). This window can be used to reduce leakage between frequency bins.

To show the real-time feedback, we want to have an abstract visualization to help the performance. There have been simple visualizations in the normal music player that frequency bands played while the music disk is playing (Play, 2016), so we design a similar frequency band as an abstract graphics in this early prototype.

Since the music visualizer should be shown in a clearer way, a music visualizer is designed: 8 cuboids are used to represent frequency bands then for each of 8 bin ranges, the columns are scaled and plotted in real-time according to the relative strength of the frequency components in each bin as HoloLens screenshot of Figure 22.
4.3.2 Onset Detection

The onset-detection algorithm implemented in this project is known as “spectral flux”. It calculates the aggregate differences for several bins between spectrum data at two close points in time (Dixon, 2006). The number of bins is set to be 1024 in this project. It means comparing spectrum data for the audio that is playing during the current “frame” to the data from the last several frames. We keep this history for comparison (Jesse, 2018). GraphMusic also calculates the difference between the most recent spectrum data and the current spectrum data per frequency bin. To clean the data, we only keep positive differences, so that we can see if we are on our way to an onset in the whole spectrum, which is called the rectified spectral flux (Jesse, 2018).

In the first iteration of GraphMusic implementation, I have run this algorithm over multiple frequency ranges simultaneously to remove noise on many different things that could be happening in an audio file at a point in time.
I also create a threshold (1 in music following function 2 and 1.5 in composition notes generation function 3) based on our spectral flux values tested for the time frame surrounding. This allows us to determine if the changes in the spectrum are significant enough for us to consider as a note (peak). We average the frame of previous spectral flux values and multiply it by sensitivity multiplier. If the spectral flux value processing is higher than the average, we can trigger the onsets and check the frequency as follows for music recognition (Jesse, 2018) in the next stage.

In summary, spectral flux is comparing the sample’s pruned spectral flux to its immediate neighbours and prepare for frequency detection.

4.3.3 Frequency Detection

I design a simple frequency detector first, but it did not get good results. Then, some extra methods are conducted to optimize this frequency detection at the second iteration by adding Infinite Impulse Response filter and change the data structure to the circular buffer.

Pitch Detector in my project is implemented by detecting the highest peak first. The high peak in the spectrum audio data has the highest possibilities to be detected as the note currently playing. The highest peak from frequency distribution in the spectrum data is extracted for each frame but it doesn’t get the accurate results at the first stage. The test is done through the check of spectral flux and pitch detection, and if they are both triggered, the system will print out the detected pitch. There is no pitch matching the accurate note range, so I continue to search for the other methods to optimize the system.

This preliminary implementation includes the step of identifying the spectrum data of isolated peaks from frequency distribution and pick out the five highest peak. Then, the one with the highest amplitude out of the five is considered as the fundamental frequency of current note.

Equation 2 Equation to get the pitch value (frequency). FreqN is the spectrum data with highest peak. Sample rate is obtained from audio setting. binSize is set as 1024.

\[ \text{pitchValue} = \text{freqN} \times \left(\frac{\text{sampleRate}}{2f}\right) / \text{binSize} \]
A pitch tracker in Unity List is an open-source library that optimizes frequency detection system and is better than the first implementation method mentioned above (tbriley, 2019). After the test and check, we can find the developer try to add Infinite Impulse Response (IIR) filter function and circular buffer data structure to the frequency detection which makes the detection more accurate (Blaise, et al., 2017). IIR filter is a linear low pass filter function of digital processing. Research shows that IIR algorithm can obviously improve the accuracy and robustness of the audio system (Blaise, et al., 2017). Circular buffer is a data structure to uses a single, fixed size buffer that is connected end-to-end. Thus, I call the frequency detection function in this library for the test. Using these 2 methods can get higher accuracy and stability according to the comparison. For each frame, we get the spectrum data and then get the sort of them to match the notes data to the dictionary.

Currently, the frequency range I set for music recognition is in 2 octaves starting from middle C for testing. We need to compare the frequency to the music note pitch frequency to decide which note currently playing. I show the following Table 3 and Table 4 of how I set frequency range according to the frequency of each music note, but frequency range is rough and set as big as possible to improve the sensitivity of the music detection. The range can be set more accurately in future work if the recognition is more accurate.

Table 3 Note and frequency setting comparison chart for the frequency detection in Octave 4 (from C4-B4) (mwguitars, 2013)

<table>
<thead>
<tr>
<th>Music Note</th>
<th>Frequency (Hz)</th>
<th>Frequency Detection Range (Hz-Hz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>C4</td>
<td>261.626</td>
<td>247-277</td>
</tr>
<tr>
<td>D4</td>
<td>293.628</td>
<td>278-311</td>
</tr>
<tr>
<td>E4</td>
<td>329.628</td>
<td>312-339</td>
</tr>
<tr>
<td>Music Note</td>
<td>Frequency</td>
<td>Frequency Detection Range (Hz-Hz)</td>
</tr>
<tr>
<td>------------</td>
<td>------------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>C5</td>
<td>523.251</td>
<td>509-554</td>
</tr>
<tr>
<td>D5</td>
<td>587.330</td>
<td>555-622</td>
</tr>
<tr>
<td>E5</td>
<td>659.255</td>
<td>623-680</td>
</tr>
<tr>
<td>F5</td>
<td>698.456</td>
<td>681-739</td>
</tr>
<tr>
<td>G5</td>
<td>783.991</td>
<td>740-830</td>
</tr>
</tbody>
</table>
After getting the highest peak, we calculate the estimated frequency in that interval and match the pitch according to the settled frequency range. After the test, larger bin length and sample rates get better accuracy results since the note can be detected smoothly as the 6.1 section self-evaluation shown.

### 4.4 AR Interactions

AR interactions in this scene are basically hand gestures and real-time audio analysis with visual feedback. The audio has been described in section 4.3 and visual feedback would be described in section 4.5. This section is mainly about hand gesture interactions.

The main menu for selecting different functions are designed (audio visualizer) in this HoloLens Application as Figure 23 shown. In the main menu, users can choose to click the menu options to enter different scenes for practicing mode (read existing music sheet), composing mode (record the music pitch), and visualizing audio mode (audio visualizer for real-time audio). The options are set “intractable” like buttons, so users can air tap (“click” gesture in the HoloLens 2) these buttons even if in a far distance (maximum 4 meters in my settings).

The navigation between the scenes is using SceneManager (an in-build Unity function) to control the scenes. If the button is triggered, Unity will unload the current scene and load the new destination scene. Hence, users can click the buttons to change among different scenes.
Implementation

Figure 23 Main menu for selecting the sub-scene in play mode

Figure 24 Main menu for selecting the “Practice” scene and show music sheet choices
In these sub-scenes, there is a back button for each sub-scene designed to come back to the main menu for users to have an experience of every other scene.

4.5 Visual Feedback

We design different visual feedback in the music tracking scene and music composition scene.

In the tracking scene, we designed visual feedbacks to show the track for users while the specific note is detected and matched. An intelligent pointer is designed as a vertical line (line renderer in Unity) overlapped on the current note and it waits to get to the next note.

After detection of the current note, the line will jump to the position of the next note automatically. The distance is in advance calculated and stored in a list while parsing the MusicXML file and drawing the 3D music sheet in section 4.2.

For a better visualization experience, this application is designed with an automatic page “scrolling” function. After the music pointer moving 2 bars, the whole music sheet will move to the left with a distance of 2 bars. The 2-bar distance is calculated by the bar horizon position difference.

In the composition scene, visual feedback is designed for showing the pitch on the sheet. If the algorithm detects one onset and the pitch is in the frequency range of octave 4 (started from middle C), the user can get the feedback of little circle marking what pitches were users playing. Similar to the last “scrolling” function, this composition scene will 0.6 meters to the left if the circles have reached the length of the 0.6 meters to right to let the users viewing the latest notes immediately shown in Figure 25.
4.6 Summary of Implementation

This chapter has described how an intelligent AR software implementation as an early prototype has been developed for users to visualise the music sheet and developed AR-Live Score to have visual interactions from the analysis of real-time audio sources automatically. Music detection technology has been used in this project and AR feedback and gestures have been implemented.
The finished software application mainly contains three functions summarized as follows. The first implementation function is to visualize the audio in an abstract way that shows frequency bands moving up and down with the beats to build a better environment for musicians’ performance as in Figure 26.

![Diagram](image)

Figure 26 The first function is to visualizes music in frequency bands in real time.

2 I have made video demo for the software application. They are all continuously videoed in HoloLens 2. The link of “Graphic Music” is https://www.youtube.com/watch?v=D4bID7gfyg,

“Play Score” is https://www.youtube.com/watch?v=FfhWoR1p5g8 and https://www.youtube.com/watch?v=jk5BopUhUHc, and “Write Score” is https://www.youtube.com/watch?v=r0ugrTho2Jw.
The second function is assisting the performance by automatically constructing a 3D music sheet from a music file and displaying this 3D music sheet in the HoloLens and then tracking users’ music playing while pointing out where they are in the music sheet as in Figure 27.

![Figure 27](image)

Figure 27 The second function displays a 3D music sheet from a MusicXML file. An audio source, such as a performing musician, can then be tracked in real time and their place in the music can be displayed.

The third function is to assist composition to display composers’ pitches in a 3D score according to the music played in real-time as in Figure 28.
Figure 28 The third function records a pitch being played in real time and visualizes this as music on the 3D.

In addition to the treatment in this chapter, Appendix C presents three paper prototypes of intelligent music systems for improvisation: (i) showing hints about the harmonising notes, (ii) creating a virtual music synthesizer and (iii) visualization of polyphonic notes detection.
5 Limitations

There are some known limitations in the implementation of this individual project. As an early prototype, there are limitations in the music sheet representation, music detection, and interactions restrictions.

5.1 Music Sheet Representation

Music representation is limited in demonstrating complicated orchestra sheet music and conventional five-line staff in suitable symbols.

At this early stage, since this AR application only contains treble, and bass clef, simple notes (eighth note, quarter note, half note and whole note) and rest symbols (eighth rest, quarter rest, half rest and whole rest) of sheet music, the sheet is limited and cannot represent music that needs accidentals, marks, notes relationship and many other music symbols. Music sheet at this stage lack symbols to represent the other complicated symbols, including different types of bar line (double line and dotted line), note relationship symbols (tie, slur, tuplet, and slur), sharp and flat, etc.

The music representation is also limited to display only one line of music at a time. Normally, musicians read several lines of music sheet at one time in the music performance. Hence, several music lines with different parts are required to develop in the future.

5.2 Music Detection

The music recognition cannot guarantee the exact accuracy in this early prototype and thus it requires improvements in the music detection algorithm. The related materials are researched and IIR and circular buffer as Implementation described.

I have attempted several methods to optimize the music detection in terms of algorithm, remote rendering, and audio settings. Remote rendering in HoloLens is to run the application on the laptop or computer. HoloLens has 4 microphones and they are detected
Limitations

and analysed together, so these steps will decrease the audio data analysis speed. Changing the audio settings to the “mono” from “studio” will be helpful since this is the music recognition algorithm to analyse the monophony.

However, there is a limitation that this microphone music detection is only supported for monophony which detects a single note at one time. It cannot detect polyphony in this algorithm for complicated chords and harmony. There are possibilities to improve the music detection in algorithm and the microphone ability as well.

Several microphones (VR devices microphone, laptop inbuilt microphone and external microphones) have been tested in this project and microphone abilities affect the quality of the detection. After testing, the Alienware laptop’s inbuilt microphone achieved the best performance in the Evaluation section. There are microphone ability limitations in my project.

5.3 Interactions

The interactions are limited now since the audio detection has occupied a great capacity of the laptop processing. The application is detecting the real-time audio, and thus HoloLens speech commands are conflicted and restricted. If we still set the speech commands when analysing the real-time audio, the spoken words cannot be detected after the test.

The other gestures’ interactions are also restricted. If the user wants to restart the music track, it is not flexible to interact with the music sheet modification on the sheet music. The user is limited in tracking from the start again without well-developed interactions.
Limitations
6 Evaluation

6.1 Self-evaluation

A self-evaluation was conducted to test the usability of this application and the sensitivity of the music recognition algorithm.

There are two parts to this self-evaluation. The first part was to undertake an audit of the extent of the music representation that had been implemented, and the second part was to evaluate the sensitivity of music recognition.

6.1.1 Music Representation Evaluation

Starting from a “music dictionary” is listed for common music symbols of conventional sheet music to check the completeness of the music representation (Dolmetsch, 2019).

Table 5 lists the audit results of the music representation as a music system for each element as selected from a “music dictionary”. Most of the basic symbols have been implemented. However, several lines as a music system are not implemented and some symbols (double bar, music start, and end, clef, key signature, and anatomy) are not implemented for this intelligent system. Also, Table 6 and Table 7 are demonstrated the completeness of notes and rest symbols respectively.

Table 5 Evaluation of sheet music elements as a sheet system (Dolmetsch, 2019)

<table>
<thead>
<tr>
<th>Music symbols</th>
<th>Music symbols name</th>
<th>Finished or not</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Staff, stave: a framework of five lines on which musical notation is written</td>
<td>Finished for 5 horizontal lines</td>
</tr>
<tr>
<td></td>
<td>Evaluation</td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>---------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>1</td>
<td>System: notation of a line of music including all the parts and voices</td>
<td>Not finished for different parts; Finished for one line part in the system</td>
</tr>
<tr>
<td>2</td>
<td>Bar line: a vertical line (or lines) drawn across a staff</td>
<td>Single bar line is finished. Double bar line is not finished.</td>
</tr>
<tr>
<td>3</td>
<td>Music start and music end</td>
<td>Not implemented (currently with only a single bar line)</td>
</tr>
<tr>
<td></td>
<td>Clef: graphical symbol on the left of the stave which establishes the relationship between particular note names and their position</td>
<td>Treble and bass clef have been set; The others are not finished.</td>
</tr>
<tr>
<td></td>
<td>The main elements of a musical score</td>
<td>Notes, time signature and treble clef are basically set, but the key signature has is not finished</td>
</tr>
<tr>
<td></td>
<td>Anatomy of a note, a single sound of a pitch and length notated with a symbol made up of a note head or a stem or a flag, with notes bearing flags are</td>
<td>Not implemented for the relationship and anatomy of the notes</td>
</tr>
</tbody>
</table>

56
Table 6 Detailed music notes evaluation (Dolmetsch, 2019)

<table>
<thead>
<tr>
<th>Music notes</th>
<th>Music symbols name</th>
<th>Finished or not</th>
</tr>
</thead>
<tbody>
<tr>
<td>🏧</td>
<td>Double whole note</td>
<td>Not implemented</td>
</tr>
<tr>
<td>⚠️</td>
<td>Whole note</td>
<td>Tested</td>
</tr>
<tr>
<td>🎵</td>
<td>Half note</td>
<td>Tested</td>
</tr>
<tr>
<td>🎵</td>
<td>Quarter note</td>
<td>Tested</td>
</tr>
<tr>
<td>🎵</td>
<td>Eighth note</td>
<td>Tested</td>
</tr>
<tr>
<td>🎵</td>
<td>Sixteenth note</td>
<td>Tested</td>
</tr>
<tr>
<td>🎵</td>
<td>Thirty-second note</td>
<td>Not implemented</td>
</tr>
<tr>
<td>🎵</td>
<td>Sixty-fourth note</td>
<td>Not implemented</td>
</tr>
</tbody>
</table>
### Evaluation

<table>
<thead>
<tr>
<th>Rest notes</th>
<th>Symbol name</th>
<th>Finished or not</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>One hundred and twenty eighth rest note</td>
<td>Not implemented</td>
</tr>
</tbody>
</table>

Table 7 Detailed rest note evaluation (Dolmetsch, 2019)

<table>
<thead>
<tr>
<th>Rest notes</th>
<th>Symbol name</th>
<th>Finished or not</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Double whole rest</td>
<td>Not implemented</td>
</tr>
<tr>
<td></td>
<td>Whole rest</td>
<td>Tested</td>
</tr>
<tr>
<td></td>
<td>Half rest</td>
<td>Tested</td>
</tr>
<tr>
<td></td>
<td>Quarter rest</td>
<td>Tested</td>
</tr>
<tr>
<td></td>
<td>Eighth rest</td>
<td>Tested</td>
</tr>
<tr>
<td></td>
<td>Sixteenth rest</td>
<td>Tested</td>
</tr>
</tbody>
</table>
Music Recognition Evaluation

Before the test on the music recognition of the AR application, we need to create the midi files as sample music. Muse Score was used to import the MIDI files and MusicXML files respectively after the same sheet music is drawn in Muse score layouts. Two famous songs were used to test the system: “Twinkle, Twinkle Little Star”, and “Pachelbel’s Canon”.

Garage band is used to edit these MIDI files for the test while the MusicXML file is used to draw out 3D music in HoloLens and extracted for the match of music.

Parameters are mainly controlled in the music recognition evaluation by GarageBand in-built music instrument styles and tempo of the music (speed). The speed of the music can reflect the music recognition sensitivity.

To keep the experiment accuracy, I set the experiment results as the fastest tempo we can get for each music clip and musical instruments. All the results are tested 3 times continuously to ensure that the HoloLens application can successfully detect each note until the end. The fastest successful tempo for each instrument and song is shown in Table 8.
Evaluation

Table 8 The evaluation of music recognition algorithm (120bpm is set as the maximum test tempo for the self-evaluation since 120bpm is relatively fast tempo).

<table>
<thead>
<tr>
<th>Tempo (BPM)</th>
<th>Little star</th>
<th>Canon</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grand Piano</td>
<td>120</td>
<td>56</td>
</tr>
<tr>
<td>Classic Electric Piano</td>
<td>120</td>
<td>120</td>
</tr>
<tr>
<td>Acoustic Guitar</td>
<td>120</td>
<td>106</td>
</tr>
<tr>
<td>String Ensemble</td>
<td>120</td>
<td>111</td>
</tr>
</tbody>
</table>

The highest tempo is set to 120BPM (beats per minute) as relatively fast for normal music tempo like a typical “march tempo”. From the table, we can see the simple music pitch such as “little star”, we can get relatively promising to get 120BPM all the time. In comparison, relatively harder and longer music fragments cannot achieve 120BPM for every musical instrument. Classic Electronic Piano gets the best results, but the Grand Piano can reach the last note only if the tempo is slower than 56BPM. The possible reason to cause differences in music instruments is that musical instruments have different timbre.

There are exceptions when the music is playing in extremely slow tempo such as that lower than 25BPM, the music cannot recognize the music successfully as this table shown. The possible reason might be the long note duration will not trigger the onset detection. Hence, extremely slow music cannot be detected by this algorithm.

Overall, this prototype can achieve normal playing speed and detect simple pitches smoothly for most of the musical instruments.

6.2 User study

A user study was also conducted in this project to evaluate if this early software prototype in music visualization (“GraphMusic”), music composition (“WriteScore”), and music
track ("PlayScore") is matched with musicians’ requirements for visualization in music performance.

There were 10 questions in the survey with 4 video links for each function. The survey link is sent through to the participants and questions are set in Survey Monkey.

There were 8 participants who took this survey. All the participants had solid music backgrounds who are musicians or current ANU music students. Generally, the participants liked the ideas of 3D music and intelligent system in AR, but some of them were not convinced that the AR system could handle complicated sheet music and complex notes detection.

6.2.1 Basic Information

This survey covers the different age group participants from 18 to over 65 years old.

First, we have several questions of the basic information to observe the normal application used by musicians. Only two participants used MusicXML, and most of the users use Ableton, Sibelius or even pdf files or written notes.

Five of the users used MuseScore in reading sheet music, and the others used Sebelius to edit music. Three of the users do not meet huge problems during performance. Two of the users pointed out they are conventional users who do not like the music software and they only use paper sheet music. One of the users has met contract agreement problems with the software developers about the specified income and music copyright issues from the music creation in the software, so this user does not like music software.

In terms of those players using the software, there are also detailed complaints about the application layouts that they not intractable and the sheets are mostly clunky and oblique in music software. Also, there are users who particularly want an accurate converter from songs to music scores, since the converter from songs and MIDI files to music scores are messy.
All the users use five-line staff for performance and improvisation, and they are comfortable about the reading conventional music sheet. Meanwhile, MIDI, Sibelius and PDF are mentioned to be used as well.

6.2.2 Music Visualizer

The questions of general AR 3D feelings of music visualizer to during the musician's performances are set in the survey. In terms of the visualization questions the following feedback was obtained:

- Half of the participants were not interested in the visualization and considered the 3D visualizer in AR headset would be distracting and disruptive.
- One of the participants suggest visualizing more accurate pitch or music data or the visualization such as rhythmic accuracy or intonation for singers and wind players could be useful in an AR environment.
- For each person there were different expectations on the AR graphical design and movement.
- Musicians will also have individual interpretations to the visualization, especially for the experienced musicians.
- But 3 participants point out that they liked the music sheets and it would be more useful in music practicing and sight reading instead of music performance. In another way, it might be potentially useful in music educational and inexperienced musicians since it is simple to read and clear to follow.
- Besides, for the dynamic changes in 3D video experience, one of the users reflected that it is dizzy to watch AR videos.

6.2.3 Music representation and tracking

3D sheet music with 3D music symbols were shown to the survey participants and this music tracking evaluation asked about the usefulness of visualization in 3D space and if 3D symbols and non-background sheet make a difference. The following feedback was obtained:

- This simple music tracking system was a positive function as an idea for the users.
• Two users described it is an eye-catching representation which was easy to follow.
• Another two participants suggested that it would only be useful when showing the chords and more complex sheet music.
• One user pointed out that improving the stability of the sheet in the scene would also help the musicians to focus on and read it better.
• Besides, another participant found there was a delay about 80ms in the real-time audio analysis in this project. The delay time length is provided from this participant, and we can test it in the future.
• In terms of the sheet music background, only 2 participants liked the transparent backgrounds with the music notations flowing in the 3D space. The other 2 participants liked a conventional feel of reading the sheet with background and one user suggested having a semi-transparent background might help. One user had no preference. The interesting part was that one participant mentioned that no background might be helpful to see the conductor and other players while reading the music sheet together.
• In terms of the tracking system, all the participants liked the idea of audio feedback and a cursor, but they also give some ideas about their expected tracking forms. Two of the users described that most of the situations, they will read at least 3 bars at one time, so 2 bars setting is slow for them. One participant gave the idea of setting a custom variable such as (1, 2, 4) by users as the Graham Fitch (Zlata, 2020) which is a pianist magazine and users can set this variable according to what speed of music they are playing. There was also a suggestion to avoid the distraction of cursor is that to differently colour the note currently playing.
• The other use of this application was suggested that one participant wanted to use this application to practice the piano or singing scales, exercise and arpeggios. This user also liked AR musical virtual objects integrating into the normal life such as waking you up or playing with you.

6.2.4 Music composition

The music composition evaluation was carried out to evaluate the usefulness of this real time pitch visualization function and if this function can help with assisting recording the
music pitch. All the participants gave completely positive feedbacks of the last composition parts in recording the pitch of the music. They all agreed that this idea was useful but also gave suggestions related to the composition components:

- Not only for the music instrument performance, but intonation and song transcription can also be used according to 3 musicians’ response.
- One participant like this application since it can free from the bulky music hardware, but this participant suspected that if AR application can handle the orchestral and choral scores.
- Also, one user suggest that the system can represent different notation such as Feldman’s projection (Boutwell, 2012) or more music information such as representation of rhythm, tempo or time signature.
- Besides, the application can also be applied to combining with other software in analysing and fields recording.
- Since the current version only shows the pitches of notes, the other music information like the rest, rhythm and the other important information and the retrieval function are still deficient.
- To optimize this current section, the backspace function is expected. The backspace can be set as deleting the notes that are not accurately detected or the music composition the musicians want to delete. The backspace can recorrect the notes and increase the interactions with the existed note in the 3D space.

6.3 Summary of Evaluation

In the evaluation part, we conducted a self-evaluation and a user study to evaluate the Live-Score application.

Considering the special situation of the COVID-19 situation, this experiment is evaluated by using videos, so the participants may have been less enthusiastic about watching videos than if they were actually wearing the AR headset to experience immersive environment.
We expect that this application can be tested and attempted by musicians again when pandemic restrictions are eased and social distancing gets back to normal.

According to the self-evaluation, main elements of music score (Five-line staff, bars, notes, bass and treble clef) have been completed. However, the complete music system for several lines and relationships of the music notes have not considered so far.

Regarding the sensitivity evaluation, this prototype can achieve normal playing speed and detect simple pitches smoothly for most of the musical instruments.

After the self-evaluation, we completed a user study with 8 participants. The participants were from different age ranges and different specialty of music fields, and the user study was sent out on the Survey Monkey website. The main ideas and feedbacks are summarized in Figure 29 below.

![Figure 29](image_url)

Figure 29 A diagram to show the process and results of the user study evaluation
According to the participants’ feedback, we can observe that this prototype represents a conventional sheet from MusicXML files that are matched to the musicians’ requirements and composition workflow. Besides, AR technology can satisfy their requirements in developing accurate music software converter and saving space from music machines. Most of the musicians represent negative feedback in (“GraphMusic”), but they liked the accurate sheet music, pitch recorder, and real-time audio analysis for composition in (“PlayScore” and “WriteScore”). Generally, they suggest providing a more accurate music audio analysis, a more completed music notation representation system, and a well-developed backspace function or modifying sheet music to interact with the AR environment.
7 Conclusion and future work

7.1 Conclusion

A music system prototype in HoloLens 2 has been designed and built. This prototype comprises a music visualizer, a music sheet visualization, a music tracking system, and a pitch recorder. This system has been evaluated by self-evaluation and a user study.

The music visualizer (“GraphMusic”) has demonstrated real-time audio detection and graphical display. The conventional sheet music visualization (“PlayScore”) uses 3D symbols to display the score with a music tracking system as its second function. We also designed a scene to show the pitches in real time for composition mode (“WriteScore”).

In the early stages of this project, an abstract player was designed to demonstrate the idea that a HoloLens AR device was “listening” to the real-time music. Next, the conventional music sheets were displayed to demonstrate the ideas of building 3D sheets and symbols in the AR headset. This prototype also supported ideas of intelligence in a tracking system for music reading and real-time pitch records function in AR. These functions are gathered in the main menu for users to interact with this system.

A self-evaluation related to the music representation completeness and music recognition sensitivity was carried out. The self-evaluation gave the results that this AR sheet can basically represent simple notes but need improvements for the complex symbols. Also, music recognition has shown that the algorithm can smoothly detect simple monophony. A user study was conducted in a survey format to gather suggestions to this system and to show that this early prototype is potentially useful. Generally, most of the musicians were distracted by the AR visualizer (“GraphMusic”), but they liked the accurate sheet music, pitch recorder, and real-time audio analysis for composition. They mainly suggested providing a more accurate music representation, more completed music notation system in AR and well-developed backspace functions or modifying sheet music to interact with the AR environment.
The intelligent music system and notation representation need to be extended to represent and track more complicated musical scores. More music symbols and data to represent several parts are also required to demonstrate according to the evaluation feedback.

7.2 Future work

The following specific future developments could be:

- Optimizing the system to improve limitations by adapting the music algorithm and developing a more interactive software in HoloLens.
- Developing a stable music recognition algorithm with a polyphony detection.
- Developing a more integrated system with different parts of the music and several lines of music to represent the more complicated sheet music is also expected, since the complex representation is more practical to be used.
- In AR environments, 3D visualizations can be designed to have more interactivity than has been demonstrated in this project so far. For instance, users can use hand gestures to grab the music symbols or backspace to change the existed music sheet through AR interactions and have the history log to record what they have composed.

The evaluation of this project has been restricted to use social distancing to evaluate using videos and surveys rather than actual user experiments. Thus, we expect more intelligent systems will be evaluated in the future and evaluated in AR headsets to provide a better experience.
Bibliography


Available at: https://www.dolmetsch.com/musicalsymbols.htm
[Accessed 13 May 2020].

Available at: https://www.youtube.com/watch?v=Tv2SavmbDVg
[Accessed 10 January 2020].


Available at: https://medium.com/giant-scam/algorithmic-beat-mapping-in-unity-real-time-audio-analysis-using-the-unity-api-6e9595823ce4
[Accessed 13 March 2020].


Available at: https://www.youtube.com/watch?v=GqHFGMy_5lc
[Accessed 13 2020].


Available at: https://musescore.com/
[Accessed 7 April 2020].

Available at: https://mwguitars.com.au/2013/04/23/matching-frequencies-to-scale-notes/
[Accessed 12 April 2020].


Available at: https://www.youtube.com/watch?v=mHk3ZiKNH48
[Accessed 4 June 2020].

Available at: https://github.com/qy-zhang/DrawSheetMusic
[Accessed 12 April 2020].


Available at: https://unitylist.com/p/k53/Pitch-Detector
[Accessed 28 May 2020].

Available at: https://docs.unity3d.com/ScriptReference/FFTWindow.html
[Accessed 12 April 2020].
   Available at: https://docs.unity3d.com/Manual/StreamingAssets.html
   [Accessed 3 June 2020].

   Available at: https://docs.unity3d.com/ScriptReference/Transform-localPosition.html
   [Accessed 25 May 2020].

   Available at: https://en.wikipedia.org/wiki/Staff_(music)
   [Accessed 13 January 2020].


Appendix A Milestone

This project is expected to involve creating a visual feedback system for musical performers using HoloLens 2. Musical notations will be displayed to the performer sourced from a musical score file format, MusicXML. The HoloLens 2 will use music information retrieval techniques, onset detection, and fundamental frequency identification to calculate where the musician is up to in a musical score.

Tasks for the project could be:

1. Review relevant research works on AR in musical performance, score display, and musical information retrieval.
2. Design and implement an AR application including dynamic score display and MIR techniques.
3. Evaluate the usability of the AR application.
Appendix B Individual Study Contract
INDEPENDENT STUDY CONTRACT
SPECIAL TOPICS

Note: Enrolment is subject to approval by the course convenor

SECTION A (Students and Supervisors)

<table>
<thead>
<tr>
<th>UnitID:</th>
<th>u6277483</th>
</tr>
</thead>
<tbody>
<tr>
<td>SURNAME:</td>
<td>Liu</td>
</tr>
<tr>
<td>FIRST NAMES:</td>
<td>Zeruo</td>
</tr>
<tr>
<td>TOPIC SUPERVISOR (may be external):</td>
<td>Henry Gardner, Charles Martin</td>
</tr>
<tr>
<td>FORMAL SUPERVISOR (if different, must be an RUSCIS academic):</td>
<td></td>
</tr>
<tr>
<td>COURSE CODE, TITLE AND UNITS:</td>
<td>COMP8755 (12 Units)</td>
</tr>
<tr>
<td>SEMESTER:</td>
<td>☒ S1 ☒ S2</td>
</tr>
<tr>
<td>YEAR:</td>
<td>2019, 2020</td>
</tr>
<tr>
<td>Topic Title:</td>
<td>Augmented Reality for Real-time Music Performance</td>
</tr>
</tbody>
</table>

LEARNING OBJECTIVES:

- Understand fundamental HCI design, audio processing and evaluation methodologies of Augmented Reality (AR) systems.
- Gain a general understanding of the frontiers of AR dev frameworks and use cases.
- Apply theories and concepts associated with effective work design to real-world application.

DESCRIPTION:
This project will involve creating a visual feedback system for musical performer using an augmented reality headset. Musical notation will be displayed to the performer sourced from a musical score file format (e.g., MusicXML), or suggestions that are generated automatically for a musical improvisation. The headset will use music information retrieval techniques such as onset detection and fundamental frequency identification to calculate where the musician is up to in a musical score or in an improvisation and provide visual feedback accordingly. Tasks for the project could be:

1. Review relevant research works on AR in musical performance, score display, and musical information retrieval.
2. Design and implement an AR application including dynamic score display and MIR techniques to identify the musicians’ place in a score or improvisation.
3. Evaluate this AR application through user studies in a real-world music context.
ASSESSMENT (evaluated by the Topic Supervisor, unless stated otherwise here)

<table>
<thead>
<tr>
<th>Assessed project components:</th>
<th>% of mark</th>
<th>Due date</th>
<th>Evaluated by</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thesis</td>
<td>45</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Artefact</td>
<td>45</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seminar</td>
<td>10</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

MEETING DATES (IF KNOWN):

STUDENT DECLARATION: I agree to fulfil the above defined contract:

...... ...... Date ...14/7/19......

SECTION B (Supervisor):

I am willing to supervise and support this proposal. I have checked the student's academic record and believe the student can fulfil this contract. If I have nominated an examiner above, I have obtained their consent (via signature below or attached email)

...... 14 July 2019

Examiner:
Name: Ben Swift Signature

REQUIRED DEPARTMENT RESOURCES:

SECTION C (Course convenor approval)

...... Date
Appendix C Artefacts

Artefacts information are all recorded in GitHub link (Please contact me if you cannot open it, since it is a private link):

https://github.com/YevenLourance/Music

This project code is started from scratch by me and I arrange the code in GitHub for the version control from Figure 30 to 32.

Figure 30 the screenshot of main scene with the coding structure on the left, scene settings at the bottom and Hierarchy structure of this scene on the right.
Figure 31 the screenshot of “Write Score” scene with the coding structure on the left, scripts listed at the bottom and Hierarchy structure of this scene on the right.

Figure 32 the screenshot of “Play Score” scene with the coding structure on the left, scenes listed at the bottom and Hierarchy structure of this scene on the right.
Appendix D Improvisation Prototypes

Apart from software design, we also give several paper prototypes that are potentially useful for music performance in terms of improvisation.

- Music Synthesizer

Music synthesizer is widely used in electronic music field for improvisation. There has been a well-designed music analogue synthesizer prototype developed by Deno in Unity Editor Layout (Kablamo, 2017). We can see similar analogue design ideas in electronic free improvisation process added with echo, oscillators, beats and filters etc. In this prototype, the designer set frequency level, different trigger for each button, volume and audio speed settings with waves showing the audio output data.

In AR, we can freely manipulate in a more appropriate interactive system without space restrictions with redundant and traditional synthesizer modes. The audio visualizer can be designed in 3D in a colourful and interactive manner to inspire the improvisation inspiration with virtual buttons and sliders. The virtual synthesizer will not take up space as real synthesizer does. Also, the design in AR can have a more interactive layout and users can change flexibly by customer settings according to individual preference.

We design a music synthesizer in AR environment as the prototype with choosing the music as bass or background. Users can choose different waveforms and set volume, echo, speed, tempo, frequency or other parameters to create different audio effects. A table with beats and notes can be set as other 2D formats with velocity and time quantize in loop and the small horizontal line represent the note data.
For different beat types, the pitch and length and location can be dragged in this AR environment by hand gestures.

- Sound Analysis

We design a model to analyse the sound for several pitch at the same time for multiple musicians and test what notes are contained in real time. In the improvisation process, users might need hear what the other collaborative musicians are playing and view in sheet music format.

Since musicians have the requirements to hear and distinguish all the collaborative musicians’ playing different music pitches at the same time to decide what to improvise in real-time, an application to detect polyphonic audio and visualize the pitches is potentially useful for musicians to improvise.
• Harmonious Representation

We can get real-time sound analysis to detect the pitch and what the notes were currently playing. Mainly in performance, musicians have main rhythm playing. Harmony has been applied in music application to give hints on users to play music and improvise. A visualization is designed in sheet music format with main rhythm in black and harmonious notes hints with other colours as the paper prototype shown. Besides, for different music instrument, different hints and layouts can be designed for users. We give the examples of the piano and string music instrument prototype.

Figure 34 an example of showing 3 highest notes at one time together for group musicians’ improvisation

Figure 35 The sheet music shows the main rhythm in black notes while red notes are hint notes
Figure 36 A piano format of showing the main rhythm in red and harmony suggestion in blue.

Figure 37 A string format of showing the hints in other points.
Appendix E Information Sheet

Participant Information Sheet

**Researcher:** This research is part of an individual project conducted by Master student Zeruo Liu and academic staff members Henry Gardner and Charles Martin in the Research School of Computer Science, College of Engineering and Computer Science, at the Australian National University.

**Project Title:** Augmented Reality for Real-time Performance

**General Outline of the Project:**

- **Description and Methodology:** The purpose of this study is to gather feedback for a designed 3-D immersive visualisation artefact for augmenting the experience of music sheet and music recognition technology. In this study, a survey will be conducted using a set of open-ended questions in order to get musicians feedback in reading music in an intelligent system and 3-D music sheets. The collected information will be further analysed and used for future designs in Augmented Reality environment. The survey will take approximately 40 minutes.

- **Participants:** Participants will be professionals, students and members of the public who are familiar with the general music field.

- **Use of Data and Feedback:** It is intended that the data will be used for a research thesis, and for the potential publication in a scientific journal. Participants can request to access the results after study completion.

**Participant Involvement:**

- **Voluntary Participation & Withdrawal:** Participation in the project is completely voluntary and you may, without any penalty, decline to take part or withdraw from the research at any time during the experiment without providing an explanation, or
refuse to answer a question. If you choose to withdraw, your data will not be analysed. Given that participation is anonymous, it will not be possible to decline to have your data included at a later date after the experiment as we will not be able to identify which data is yours.

- **What does participation in the research entail?**

The participants will be asked to watch an online video and fill in the survey. During the user study, the users will see video link and go through to answer the prepared questionnaires. We may contact you for clarification on any of the information about the survey or detailed ideas in the questionnaires.

- **Location and Duration:** The study will take place anywhere safe and quite according to the users’ needs. Users can choose a safe and quiet place with a computer and well-connected network to see the video and submit the survey after the evaluation.

- **Remuneration:** No remuneration is being offered for your participation in these surveys.

- **Risks:** There are no known risks, discomforts, hazards or side effects from participation. The study can be taken place at any place for participants to be as comfortable as possible. The only discomfort in this user study is tiresome, so users can cease the user study or take a break at any time during the participation time.

- **Benefits:** The outcomes of these surveys will be used to evaluate the prototype design and the intelligent music system to enable the immersive visualisation artefact for augmenting experience of music reading.

**Confidentiality:**

- **Confidentiality:** The data from the study will be anonymised so that no participant will be able to be identified from any data collected. All results published will be in regard to the overall findings from the cohort of participants and not on an individual basis. Up until three months after the evaluation, if you give your permission, your contact details will be retained for follow-up questions.
may be used in follow-up research by researchers not listed on this form. All researchers who will gain access to the data collected in this research will be listed under the same human ethics protocol as the current researcher.

**Privacy Notice:**

In collecting your personal information within this research, the ANU must comply with the Privacy Act 1988. The ANU Privacy Policy is available at [https://policies.anu.edu.au/ppl/document/ANUP_010007](https://policies.anu.edu.au/ppl/document/ANUP_010007) and it contains information about how a person can:

- Access or seek correction to their personal information;
- Complain about a breach of an Australian Privacy Principle by ANU, and how ANU will handle the complaint.

**Data Storage:**

- Data collected from these evaluations will be stored securely in the Research School of Computer Science, ANU, and destroyed after three months. Anonymised data from the experiment will be stored securely in the Research School of Computer Science, ANU.
- If a research publication results from this work, anonymised data will be stored for a minimum of 5 years following the date of any publication. This publication data will be kept in secure storage at the Research School of Computer Science, ANU.

**Queries and Concerns:**

**Contact Details for More Information:** For further requests for information or queries regarding the study, please contact

Zeruo Liu. Email: Zeruo.Liu@anu.edu.au

Henry Gardner. Email: Henry.Gardner@anu.edu.au or

Charles Martin. Email: Chales.Martin@anu.edu.au
Ethics Committee Clearance:

The ethical aspects of this research have been approved by the ANU Human Research Ethics Committee (Protocol 2016/156). If you have any concerns or complaints about how this research has been conducted, please contact:

Ethics Manager
The ANU Human Research Ethics Committee
The Australian National University
Telephone: +61 2 6125 3427
Email: Human.Ethics.Office@anu.edu.au
Appendix F Consent form

WRITTEN CONSENT for Participants

Augmented Reality for real-time performance

I have read and understood the Information Sheet you have given me about the research project, and I have had any questions and concerns about the project (listed here addressed to my satisfaction.

I agree to participate in the project. YES ☐ NO ☐
I agree to this survey results are recorded YES ☐ NO ☐
I agree to be identified in the following way within research outputs:
Full name YES ☐ NO ☐
Pseudonym YES ☐ NO ☐
No attribution YES ☐ NO ☐

Signature: ..................................................

Date: ..................................................
Appendix G Questionnaire

I have created an Augmented Reality (AR) interactive environment to assist musicians. This system uses a Microsoft HoloLens to add a 3D graphical score to a musicians’ view and to detect the notes that they play on their musical instrument. The HoloLens allows a user to see the virtual and real world together by projecting objects on a transparent visor.

This questionnaire is to gather information from musicians and other users about how they might use this system during practice and performance sessions.

In this study you will watch a video (link) made in HoloLens, and then answer questions about how you might use this system.

During this video you will see three functions potential use cases for my prototype system:

1. Sound Visualisation: Display a 3D music visualizer to build a better performance background for musicians.
2. Performance Assistant: Drawing 3D musical score in user’s visual field. The system can track the user’s playing and point to where they are in the score.
3. Composition Assistant: The system tracks the user’s performance and adds notes that they play to a new 3D score.

Please watch the videos before answering these questions.

Interview Questions:

1. Basic Questions
   a. What musical instrument do you play?
   b. What music style do you prefer to play (classical, electronic music)?
   c. What is your age range?
      i. 15-30
2. Do you use any software to read sheet music, write compositions or assist with improvisation? If so, what software?

3. Do you encounter any problems or frustrations when using music performance and composition software?

4. What kind of music symbol have you used before? (e.g., Five-line staff, six-line staff, piano rolls…)

5. What formats do you use to store, create, and view musical scores? (e.g., PDF, MIDI, Sibelius, Finale, Ableton Live, MusicXML).

6. What do you think about the audio visualizer in the video? Does it help you have a better environment to play music?

7. In the video, the musical notes are represented in standard notation with 3D models. Try to imagine the scenario when you are playing music and music is followed smoothly. What do you think of this 3D representation of music?

8. Do you prefer to have the music sheet background like white paper or floating music sheet without background as the video shown?
9. In the video, the music pointer which is a vertical line moves as the user plays the piece of music. Do you think the playback cursor is useful? Do you think the pace of moving 2 bars each time is suitable? Have you used or considered how software could assist you while playing music from a score?

10. The system records the note pitch that the user plays. Do you think this would be useful? Are there other ways that this system could assist you to write down music as you play it?