PIANOW - PIANO SELF-LEARNING ASSISTANT IN MIXED REALITY
INTERIM REPORT

CREATED BY

BUI, THANH TUNG (UID:3035124491)
TSAI, YI-TING (UID:3035124520)

SUPERVISOR

DR. DIRK SCHNIEDERS

The University of Hong Kong

Abstract

Learning to play the piano is a difficult process that requires constant and rigorous practice. In helping new learners picking up the instrument faster, the idea of using falling notes and colored keys has proven to be useful and has been widely adapted by many piano tutorial applications. In this project, this great idea will be brought into Mixed Reality, a combination that is expected to tremendously improve the piano learning experience, as learners can follow the tutorials without looking away from their own physical piano keyboard. The new application, named Pianow, aims to offer beginners a new way to practice playing the piano by themselves, which is easy, fun, and interactive. Pianow is developed for the HoloLens. It can effectively align virtual assets to the physical piano keyboard and display interactive piano tutorials consisting of falling notes and colored keys. This interim report will discuss in the significance of Pianow in assisting and motivating piano learners and justify why the idea of falling notes and colored keys are chosen. In the methodology section, the solutions to some technical challenges, especially hologram alignment, will be presented. Finally, the current progress and the future plan of the project will be reported along with some difficulties encountered and schedule of the next phase.

Acknowledgement

We would like to express our sincere appreciation to Dr. Dirk Schnieders, our supervisor, for having offered us the best support from the very beginning of this project to now. We would also like to thank our second examiner Dr. Loretta Choi for the valuable advice she gave us. Last but not least, we would like to thank the Department of Computer Science HKU for providing us with the funding and materials needed to carry out this project. This project would not have been possible without the help from all of you.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Background & Problem Statement

The piano keyboard is composed of only black and white color. The pattern of black and white keys on the piano is repeated every 12 notes. This includes 7 white keys and 5 black keys. The reason for this color scheme is that the black keys help distinguish flats and sharps in musical notation and allow the players to quickly determine where they are on the keyboard [1].

Mastering the piano requires regular and rigorous practice. Professional pianists normally practice 6-8 hours per day and have to take regular breaks [2]. For most piano students, a reasonable amount of practice is around 30 minutes per day [2]. Although, learning to play the piano is time-consuming and repetitive, there are a lot of benefits to playing the piano. Since learners have to quickly differentiate repetitive groups of black and white keys, they acquire the ability to split concentration, which is good for multitasking in real-life situations [3]. Each hand performs entirely different movements, which stimulate multiple parts of the brain. This process is beneficial to hand/eye coordination and also inspires creativity [3].

However, not everyone can attend piano lessons regularly. A 30-minute piano lesson costs USD30-60 on average [4]. Learning to play the piano by oneself is not an easy process, especially for absolute beginners who are not able to read the music sheets. An estimation by CMUSE [5] indicates that 85 percent of piano learners give up on learning at an early stage because they dislike learning the basics. For starters, it is easy to get disheartened if the progress comes slowly.

There have been many attempts to address this problem. The most common form of assistance is self-learning tutorials by means of a video, a book or an application. All these forms, however, have their critical shortcomings. While books are not visually friendly, video tutorials are not interactive enough. From time to time, learners have to alternate between playing the piano and watching the tutorial, which causes great inconvenience. Mobile applications such as Perfect Piano are more interactive, but they fail to offer such experience on a physical piano keyboard. As a result, the learning experience on one of these applications is drastically different from that on a real piano.

Mixed Reality is an emerging technology that embraces the interactions between the virtual and the physical worlds. Many studies have shown that the application of Mixed Reality in education can significantly boost the effectiveness of teaching as well as the performance of learners. For example, Dickey [6] mentions that the use of virtual content in education facilitates a constructivist learning environment, where learners can proactively participate in practical learning activities rather than just trying to absorb theoretical concepts. Dede [7] also points out that the interactivity brought about by Mixed Reality technology allows knowledge to be obtained through hands-on practices in a virtual environment, which is very difficult to achieve in the real world.

The potentials of Mixed Reality is especially apparent when it comes to learning activities that demand constant practice and high level of interactivity like learning to play the piano. A piano tutorial built in Mixed Reality is expected to offer strong immersive experience and significantly improve the learning outcome.

1.2 Current Solutions

There are several piano self-learning tools that are available on the market. The most prominent solutions include Synthesia, Teomirn and Music Everywhere.

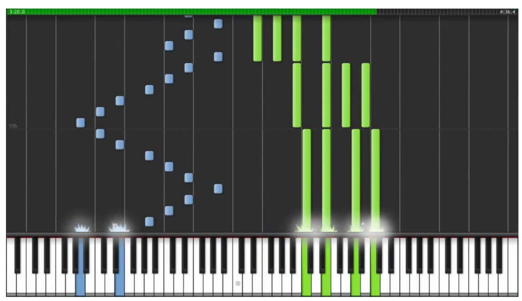

Figure 1: Synthesia piano tutorial[1]

Synthesia [8] attracts special attention due to its successful idea of building a piano tutorial using falling notes and colored keys (see Figure 1). The falling notes match with the keys and let the user know beforehand which notes to be played next. The colored keys simply indicate which keys should be pressed at the moment. The application allows the user to connect to a MIDI piano keyboard, the data from which is then passed back to Synthesia to determine whether a key has been pressed by the user and for how long it has been pressed. With the appropriate piano keyboard, the key can be illuminated to create the effect of key coloring. Thanks to its intuitive solution, Synthesia has made playing the piano easier and more fun to many people. The major downside of Synthesia is that most of the tutorial contents are still displayed on the computer screen, and the application needs to be paired with certain types of digital piano keyboard to enable all functionality.

[1]Youtube Image. Available: `https://i.ytimg.com/vi/m5RmYEd1N9I/maxresdefault.jpg`

Figure 2: Teomirn's piano tutorial with virtual hands and keyboard[2]

Teomirn [9] is a Mixed Reality application on HoloLens that teaches the user to play the piano. The design philosophy of this application is to allow the user to imitate piano experts by watching them play. The application supports 2 learning modes: watching a virtual piano expert playing on the physical piano keyboard or playing alongside a virtual hand that appears on a virtual keyboard right above the physical keyboard. Figure 2 illustrates how the second learning mode is designed. The aim of this application is to let the learner observe how a certain piece of music is played by an expert, from which they can learn the correct hand placements, movements, and the best practices when playing the piano. However, with this approach, the user is still forced to divert his attention to the tutorial. Teomirn's solution, although developed for Mixed Reality, can simply be considered a more realistic version of a video piano tutorial.

Figure 3: Music Everywhere's piano tutorial[3]

Music Everywhere [10] is another application in Mixed Reality that describes itself as a piano improvisation learning system. This system aims at providing the user with piano improvisation skill, which is not often taught but is needed when playing with other artists or in a band. Music Everywhere, hence, addresses a slightly different problem compared to that mentioned in Section 1.1. There are two main components of instruction used in this application: a virtual band along with which the user can play, and a virtual panel placed perpendicular to the piano keyboard to indicate which

[2]Youtube image. Available: `https://www.youtube.com/watch?v=aovJh2SxDYU`
[3]Twitter Image. Available: `https://pbs.twimg.com/media/C23OcqvXUAQJG-o.jpg`

keys to be pressed or for how long it should be pressed (see Figure 3). Like Teomirn, Music Everywhere does not overlay the tutorial content directly on the physical piano keyboard, which may present a significant obstacle for absolute beginners. The specific focus on piano improvisation skill also renders this application as an inappropriate choice for many piano learners. Additionally, Music Everywhere requires the use of a marker to locate the position of the piano keyboard. In the long term, this is not the best approach to be used on the HoloLens.

1.3 Motivation

The project team is motivated to provide a solution that solves the problems of self-learning piano mentioned in Section 1.1, which is better than or is a complement to the current solutions mentioned in Section 1.2. As Mixed Reality is foreseen to be popular in the near future, a Mixed Reality application is likely to cost less than hiring a human instructor. Using Mixed Reality as the medium of instruction makes the tutorials easy and intuitive, even for beginners who have no sight reading skill. At the same time, a Mixed Reality application should be fun and interactive, which will inspire beginners to keep practicing as they can see their own progress. Unlike Music Everywhere and Teomirn, however, the new application should append all tutorial content directly on the piano keyboard to make full use of the immersitivity of Mixed Reality.

1.4 Objectives

The project team aspires to develop a new Mixed Reality application to assist piano self-learning. The application is named Pianow. Pianow mainly aims to help beginners find playing the piano fun and easy at the early stage so that they have the motivation to keep learning and moving on to later stages. It can also be entertaining even for experienced piano players. In the long term, Pianow is expected to change the way self-taught pianists learn, and allow more people to feel the excitement of playing the piano. There are 2 objectives to be achieved for this application:

1. The application can recognize any 88-key keyboard with computer vision technology.

2. The application offers effective tutorials, intuitive user interface and good user experience.

The first objective involves the main technical challenge of the project. It is important that the application has knowledge about where the physical piano keyboard is located in the environment so that virtual contents can be placed appropriately. The second objective is about the practicality of the project, i.e. whether the application is effective in assisting piano learners and whether it offers good experience to the user. These two aspects are identified as the two most fundamental issues that the project needs to address in order to achieve success.

1.5 Design for Effectiveness

Pianow's two methods of instruction are determined to be falling notes and color keys. The choice of these two method are based on several factors.

The first factor that justifies this approach is the success of Synthesia itself. Synthesia uses the same idea of falling notes and colored keys as the instruction medium, which has proven to be useful for many users, based on multiple ratings and review on App Store and Google Play Store. On Google Play Store, Synthesia has more than a million downloads and holds a rating of 3.8 stars [11]. Meanwhile, on App Store, it is rated at 4.5 stars [12]. The reviews are generally positive, and a large number of users believe that Synthesia has helped them improve their piano skills in a fun and interactive way.

Nevertheless, from time to time, the most commonly mentioned issue of Synthesia-style tutorial is the fact that it discourages learners from looking at the music sheet, which will form a bad habit in the long term. However, while following the music sheet is one way to learn the piano, learning-by-ear is another method that has proven to be helpful to many people, including professional pianists. This method, proposed by Shinichi Suzuki in the 1930s, is also referred to as the mother-tongue approach, which is advocated by many influential musicians and has become widely accepted in North America [13]. The method involves 2 steps. First, to familiarize the learner with a piece of music by letting them listen to it repeatedly for many times and, second, to let them find out by themselves how to play the piece of music on a keyboard without any printed music sheet [13]. The music sheet will be introduced at a later stage, when the learner's ears have been well-trained [13]. The design of Pianow does not strictly follow this framework because it is hard to guarantee that users have been familiar with a piece of music before they start learning it, and when following the tutorial, users do not need to find out by themselves which key to press to replicate the memorized melody. Pianow, however, do embrace the idea of intuitive music learning, which allows new learners to learn music in a natural way, without too much emphasis on learning to read the music sheet.

Last but not least, by overlaying the graphical content directly on the piano keyboard, Pianow is likely to boost the productivity of the learning process. The Social Science Research Network states that 65% of human beings are visual learners [14]. Using visual instructions directly on the piano keyboard will probably allow many learners to remember the songs much faster compared to other methods.

Certainly, the effectiveness of this design needs to be formally examined. Case study, expert consultation and user-assisted testing are the intended methods for gathering related information and opinions. In the long term, it is necessary to perform an evaluation of real-life use cases over a long period of time to objectively determine the effectiveness of the application. This requires more resources and is out of the scope of the project at the current stage.

1.6 Flow of Events

The application's workflow is described as follows:

1. The user wears the HoloLens and stands in front of the piano, from a distance where the camera could view the entire piano keyboard.

2. Pianow recognizes the piano keyboard and shows a white arrow to indicate the center position of the keyboard.

3. The user uses tap gesture to display the virtual piano keyboard perpendicular to the physical piano.

4. The user sits down comfortably in front of the piano and check if the alignment of black keys in the virtual keyboard and the physical keyboard is good.

5. If the alignment is satisfactory the user chooses to move on to the next step.

6. If the alignment is not satisfactory, there are 2 ways to improve it:

- The user press "track again" button to recognize the center position of the piano again
- The user manually aligns the virtual piano keyboard with the physical piano keyboard by drag-and-drop.

7. After a satisfactory alignment is achieved, the user chooses to move on to the next step.

8. Pianow displays the control menu, which allows the user to:

- Choose a song from a list of tutorials
- Customize the speed at which the tutorial will be played
- Disable/enable the built-in music of the tutorial or an instructing feature of the tutorial (i.e. falling notes or colored key)
- Choose which hand to practice. For example, if left hand is chosen, the notes will be played automatically for the right hand and the user will play left-hand notes by himself/herself

9. The user finishes his/her customization and chooses to start the tutorial.

10. Pianow starts playing the tutorial with falling notes and colored keys. If there is Bluetooth connection, the tutorial will also provide feedback to the user on whether he/she is playing correctly.

1.7 Scope

Pianow is a Mixed Reality application implemented for HoloLens only. The piano keyboard to be recognized should be a standard keyboard of 88 keys and should be majorly black in color, with the exception of the white keys. The application only handles the recognition under stable and reasonable lighting condition (50-100 lux). At this stage, two main methods to guide user would be falling notes and colored keys. Feedback to user is only available if the piano keyboard is equipped with Bluetooth capability.

2 Methodology

2.1 System & Equipment Setup

This project requires a range of hardware and software that should be set up to work in coordination with each other during the development process.

2.1.1 Hardware

The major pieces of hardware used in this project are the HoloLens and a standard 88-key piano keyboard. The piano keyboard can be a traditional or electrical piano keyboard. It must have 88 keys and must be black in color. If the piano keyboard is Bluetooth-enabled, it can be connected to the HoloLens via Bluetooth. The HoloLens is a powerful headset designed exclusively for Mixed Reality, and arguably the best of this category on the market. It is also the most widely known Mixed Reality headset with a large supporting community. Hence, it is the most viable option for this project.

Besides, in this project, most of the development tasks are performed in a system with 64-bit dual-core CPU, 8GB of RAM and integrated Intel HD Graphics.

2.1.2 Software

For Mixed Reality development on HoloLens, Microsoft has provided a detailed list of software requirements[4]. According to this list, Unity is required as the officially supported development platform, and Visual Studio has to be used alongside Unity as the debugging and deployment tool for HoloLens.

During the first phase of the project, OpenCV is used for image processing and piano keyboard recognition. Since the piano keyboard's form, shape and features are unchanged, traditional computer vision algorithms implemented in OpenCV are sufficient for the recognition task of this project. However, as OpenCV is not officially supported by Unity and HoloLens, some integration steps need to be performed.

[4]More information available at: `https://developer.microsoft.com/en-us/windows/mixed-reality/install_the_tools`

First, the code for piano keyboard recognition is written in C++ and then compiled to a Dynamically-linked Library (DLL) file for Universal Windows Platform. OpenCV source code also needs to be slightly modified to remove some dependencies on the Windows Desktop Platform before being compiled to DLLs. This modified version is available on the opencv-hololens repository by Sylvain Prevost[5]. Finally, all the DLLs are then imported to Unity as native plugins, which can be gracefully deployed and run in the HoloLens.

Additionally, to stream data from the locatable camera of the HoloLens to Unity, the HoloLensCameraStream plugin written by Vulcan Technologies is used[6]. This plugin is necessary since the HoloLens API does not offer a native method to access the camera video stream in real-time. The plugin allows the camera stream to be played at 30 frames per second (fps), which is sufficient for the recognition task.

Finally, some functions that can be implemented using HoloLens API are adapted from the MixedRealityToolkit-Unity repository written by Microsoft[7]. This repository is a collection of pre-made components that can be utilized to accelerate the application development process on the HoloLens. In this project, the features that are adapted from this repository include input handling, spatial mapping and draggable hologram.

2.2 Implementation

This section describes in detail solutions to some major technical challenges of the project.

2.2.1 Image Processing & Piano Keyboard Recognition

The piano keyboard recognition algorithm is written in OpenCV and is based on a markerless tracking method proposed by a research group from Nanjing University [15]. The algorithm follows 3 major steps:

Preprocessing: The purpose of this preprocessing step is to extract only relevant information from the image to reduce computational cost in later steps. The input frame is first binarized with a threshold of 200, taking advantage of the fact that the piano keyboard is only composed of black and white color (see Figure 4). This threshold is chosen empirically, considering how much irrelevant details in the image can be filtered out, leaving only the keyboard area.

After that, a morphological closing operation with the kernel of size 5x5 is performed on the binary image to eliminate gaps between different keys, which ensures that the entire keyboard area is considered as a connected component (see Figure 5).

[5]Source code: `https://github.com/sylvain-prevost/opencv-hololens`
[6]Source code: `https://github.com/VulcanTechnologies/HoloLensCameraStream`
[7]Source code: `https://github.com/Microsoft/MixedRealityToolkit-Unity`

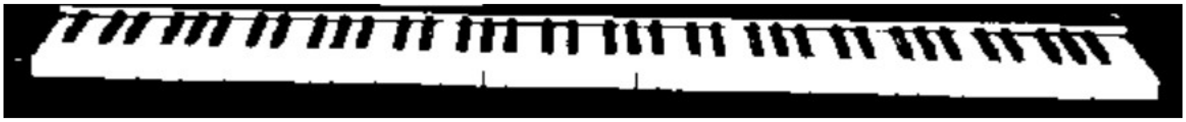

Figure 4: Piano keyboard after binarization

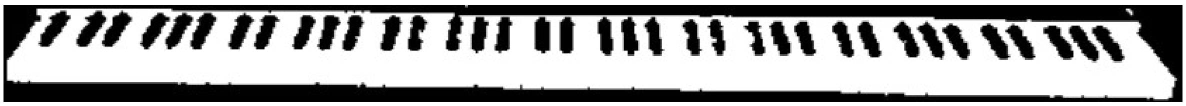

Figure 5: Piano keyboard after morphological closing operation

Contour extraction and analysis: This step is aimed at identifying the keyboard area in the image. The preprocessed image is used for exterior contour extraction. The contours are then filtered in two ways:

- The minimum-size rectangle enclosing each contour is constructed. Only the contours whose minimum-size rectangle' size and dimension are similar to that of a piano keyboard will be retained. The contours that are too small, too short or have unmatched dimension will be discarded.
- The contours are then approximated using Convex Hull and Douglas-Peucker approximation. Convex Hull is meant for including all black keys in the keyboard area, while Douglas-Peucker approximation is intended to approximate the contour containing the piano keyboard to a quadrangle, using an appropriate epsilon value. In this case, the epsilon value is empirically determined to be $\epsilon = 0.015P$, where P is the contour perimeter. Only the convex hulls of those contours that can be approximated to a quadrangle will be retained.

After the filtering steps, if there are more than one remaining contour, one will be chosen as the piano keyboard area. On the other hand, if no contour is retained, it is concluded that the keyboard does not appear in the current frame.

Obtaining the keyboard's position and orientation: The keyboard position is determined by its center of gravity (point A in Figure 6), which can be calculate using OpenCV Moments class. The center of gravity is chosen because it proves to be more stable and consistent compared to the geometrical center, especially for input frames of low resolution. Next, the the eigenvector of the contour is calculated using Principal Component Analysis (PCA) (vector AB in Figure 6). This eigenvector indicates the orientation of the keyboard. After the algorithm finishes processing a frame, the coordinates of two points A and B will be returned to the main program.

Figure 6: Convex hull (blue), center of gravity (pink) and eigenvector (green) of the piano keyboard

2.2.2 Pose Estimation

In the main program, the coordinates of A and B are first translated from camera coordinates to world coordinates. This function requires the knowledge of the camera-to-world translation matrix and projection matrix of the camera. These matrices can be obtained using the HoloLens API, and the translation procedure has been implemented in the HololensCameraStream plugin. After the world coordinates of A and B have been calculated, two raycasts are projected from the camera to A and B, which hit the spatial mapping mesh of the piano keyboard at C and D respectively (see Figure 7). C is considered the center of the piano keyboard and vector CD indicates the orientation of the keyboard in world space. With this information, a hologram can be easily placed on top of the physical piano keyboard by modifying its transform's position and rotation properties.

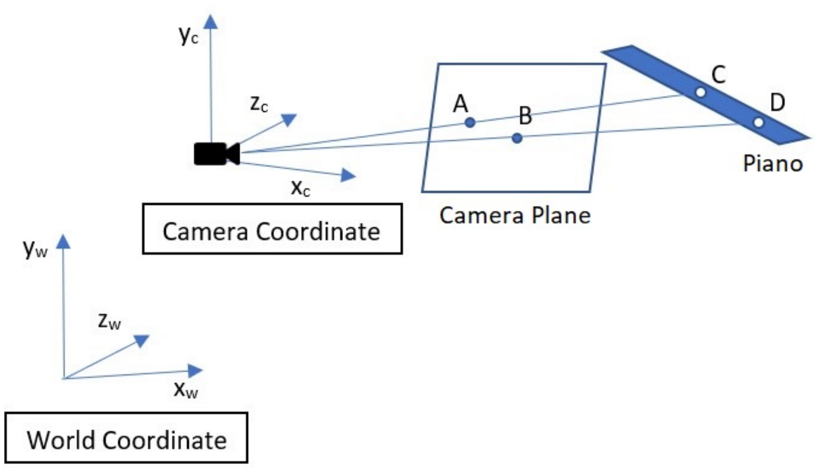

Figure 7: Pose estimation

Appropriate offsets are added to the transform (see Figure 8) as the holograms rendered by the HoloLens are not perfectly static and will shift their position slightly as the user moves closer to the piano keyboard after the recognition is done.

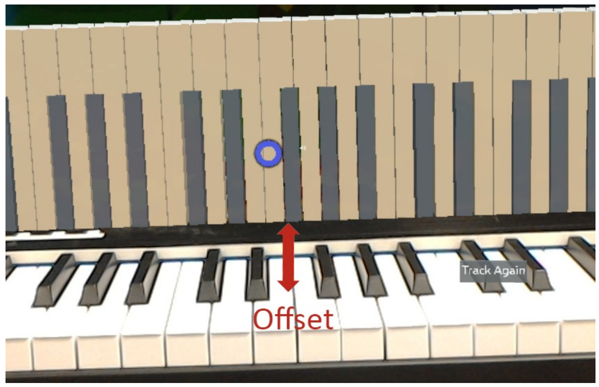

Figure 8: Offset of the hologram relative to the piano keyboard

2.2.3 Manual Alignment

Due to the unstable quality of the locatable camera stream and the spatial mapping mesh produced by the HoloLens, the recognition result is not guaranteed to be always accurate. To ensure that a good alignment can always be achieved, the user is allowed to perform manual alignment by drag-and-drop. To make the process easier and more intuitive, a 3D model of the piano keyboard, the size of which accurately matches that of a physical piano keyboard, is prepared. The model is attached with HandDraggable script from HoloToolkit, allowing it to be dragged and dropped by the user. The model is oriented perpendicular to the physical keyboard to make it easier for the user to see both the virtual and the physical keyboard and align them with each other. Additionally, a separate layer mask is assigned to the keyboard model in Unity. This layer mask is is prioritized to receive raycast hits to ensure that the model can always be selected for drag-and-drop.

2.2.4 User Interface & Tutorial Content

The User Interface (UI) of the application will mostly be adapted from pre-made assets in HoloToolkit bundle, with appropriate modifications to suit the project. The design of the UI is kept simple with only one main control menu from which the user can perform all actions (see Figure 9).

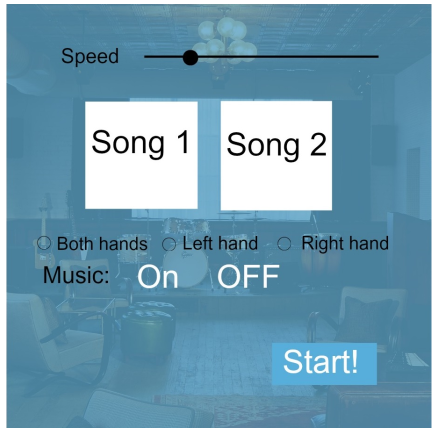

Figure 9: Control menu design (draft)

For proof-of-concept purpose, at this stage, the tutorial presentation will be made in the form of video tutorials, which can be scaled to fit the physical piano keyboard and inserted to Unity scene as media objects. There are video tutorials readily available online that can be used with appropriate permissions. In the long term, the process of tutorial creation can be automated, given appropriate data for synchronization such as which keys to be pressed and how long it should be pressed at a certain time.

The colored keys can be implemented by adding a hidden piano keyboard model perfectly overlaying the physical keyboard model. At each time a key in the physical keyboard is supposed to be colored, the corresponding key in the overlaying model will show up on top, having the desired color.

To enhance interactivity, the application can be connected to a Bluetooth-enabled piano keyboard. The MIDI data will be communicated from the piano keyboard to the application, which will allow the application to provide timely feedback to the user.

2.3 Testing & Evaluation

To assure the quality of the application, a criterion of success and a testing method have been assigned to each component of the project (see Table 1). To facilitate the testing process, a group of test users is selected, preferably at the size of at least 10 people. Test users may or may not have previous experience in playing the piano. Detailed testing method will be devised at a later stage.

Criterion of Success	Testing Method
The application should be able to recognize any 88-key piano keyboard under reasonable lighting conditions. Good alignment should be achieved easily.	Test the recognition and alignment mechanism under various lighting conditions (50-100 lux), using different standard 88-key piano keyboards. Moving the head by at most 10cm in all directions and check if the holograms are still aligned correctly with the physical keyboard. A good alignment should have an offset of less than 3mm.
The user interface and application workflow should be easy, intuitive and satisfying.	Based on observations of test use cases and feedback from test users.
Tutorial content should be appropriate and effective.	Consult expert opinions as well as feedback from test users. Observe the progress made by test users.

Table 1: Testing criteria and methods

3 Interim Result & Evaluation

3.1 Work Accomplished & Evaluation

At this stage, the first objective of the project has been achieved. When the user wears the HoloLens and look at the piano keyboard, a white arrow will appear to indicate the center of the keyboard (see Figure 10). The user may need to adjust his/her own perspective of the physical piano keyboard (e.g. by moving closer or further) to achieve the best recognition result. Generally, the best result is achieved when the user faces the piano keyboard upfront, and the locatable camera can capture the entire keyboard. When the lighting condition is not ideal (e.g. too bright or too dark), the arrow pointer tends to be less stable, but a good recognition result can still be achieved eventually.

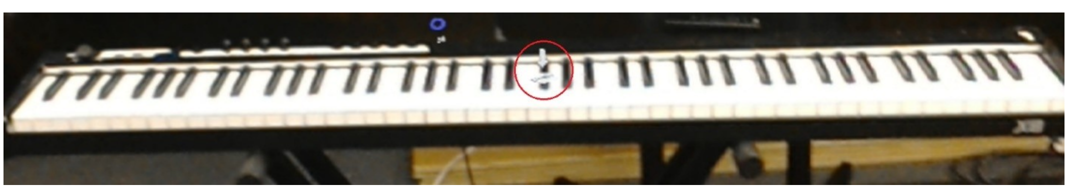

Figure 10: Keyboard center marked by a white arrow

After obtaining a stable and well-positioned indicator, the user can perform a tap gesture to confirm the position of the keyboard. A 3D model of the piano keyboard will be rendered at that position, and its orientation should match that of the physical piano keyboard (see Figure 11). The 3D model is not closely attached to the physical piano keyboard due to some programmed offset values (refer to Section 2.2.2) . This is because when the user moves closer to the keyboard, the hologram is expected to shift its position and rest right above the top edge of the keyboard. At this point, the user can evaluate the alignment result by matching the black and white keys on the 3D model with those on the physical keyboard (see Figure 12). The position and orientation of the hologram becomes more accurate as the HoloLens has more time to scan the environment and obtain a better spatial mapping mesh. Generally, the success rate is of around 70%.

Figure 11: Hologram rendered on top of the piano keyboard

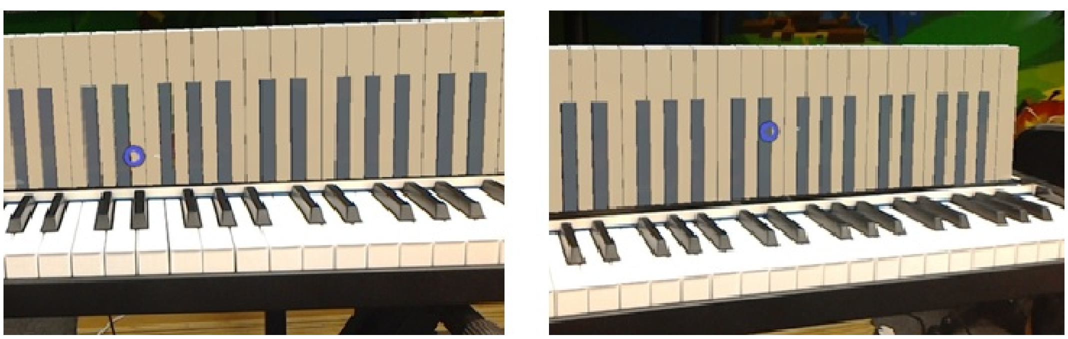

Figure 12: Alignment result: low accuracy (left), high accuracy (right)

Even though the recognition accuracy is not guaranteed, the user can use simple gestures to drag and drop the hologram and change its position (see Figure 13). This process is easy and intuitive for any experienced HoloLens user. It also allows the placement of the hologram with high accuracy and flexible choice of location.

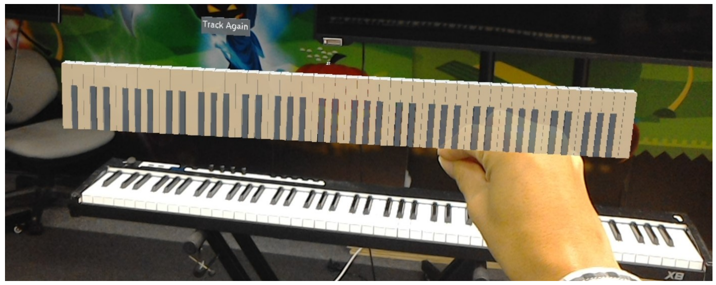

Figure 13: Drag-and-drop for hologram alignment

After a good alignment has been achieved, if the user does not move away too much from his/her original perspective, the alignment result will be maintained. The tolerable movement distance of the head is roughly 10 centimeters to all directions. If the user finds that the alignment is not sufficiently accurate, he/she can easily go back to any step of the alignment process to make adjustments.

Overall, the result achieved is satisfactory, and it allows later development process to be carried out as originally planned.

3.2 Difficulties Encountered

There are 3 major difficulties encountered in this project so far.

Firstly, a major difficulty was caused by the project team's lack of experience with most of the tools, such as Unity, .NET development, HoloLens API, OpenCV, etc. This introduces a big learning curve to every development steps and requires extensive effort and patience in researching and debugging.

Secondly, most of the tools related to the HoloLens are new. Therefore, the documentations are not detailed and well-organized enough. The community is also highly immature, so there is not much support if any problem is encountered during the development process. Additionally, since the technologies are still under development, there are many shortcomings. For example, different versions of Unity may cause conflicts with different versions of other software, so it is difficult to decide on a version that can integrate gracefully with every tool. Besides, as the HoloLens' locatable camera cannot be accessed from the Unity Editor, it is impossible to test the application within the editor environment. The consequence is that every testing and debugging process requires deployment to the HoloLens, which significantly slow down the development progress.

Finally, the HoloLens itself has many unexpected limitations that cause the project's approach to change several times. For instance, the video stream from the HoloLens' locatable camera has a much lower resolution than how it is claimed to be. This makes it impossible to achieve a very good recognition result and requires additional alignment method such as manual alignment by the user. Other limitations such as the low frame rate, small field of view and inaccurate spatial mapping mesh also disallowed many solutions that the project team came up with.

4 Future Plan

4.1 Work Remaining

Followed from the current status, the first objective is mostly achieved. The focus of the next stage will be improving the recognition mechanism and implementing the user interface and tutorial contents.

The current recognition mechanism will be tested in different scenarios, under different lighting conditions to ensure its robustness and stability. Although the original aim was to develop a markerless recognition mechanism, the method of using a marker will still be attempted to make a comparison with the current recognition method. To better evaluate the accuracy of the alignment, some keys on the physical keyboard will be directly colored, from which the offset can be observed more easily.

Although the instruction media are set to be falling notes and colored key, further evaluation is needed to confirm the effectiveness of this approach. To make decision on the final design, the project team will consult lecturers and students from the Department of Music. The test users will also be involved in a short study to evaluate the likely effectiveness of the application. In the long term, a formal study needs to be carried out, which involves and large number of participants over a long period of time to objectively determine the extent to which the application can assist people in self-learning the piano. This is a resource-consuming process and is out of the scope of the project at this stage.

After the final design is determined, the tutorial contents and the user interface will be implemented. The implementation should follow what has been discussed in Section 2.2.4. Finally, the testing procedure will be carried out according to the discussion in Section 2.3. The procedure will take 3-4 weeks.

4.2 Schedule

Below is the schedule of remaining tasks (Table 2). We will first finalize our design after consulting some experts and move on to implementation of user interface and tutorials development from February to March. Our ideal time to complete the application is mid March, so that there will be enough time for user testing and feedback collection. On April and May 2018, the required deliverables will be finalized and submitted. The project will be concluded by the end of May 2018. Words in **bold** indicate hurdle requirements.

Time	Deliverables
Mid January - Mid February	- Finalize the application design according to input from experts - Implement control menu and instruction page - Test the recognition mechanism under different conditions
Mid February - Mid March	- Implement tutorial contents - Complete the application
End of March	- Quality assurance test and feedback collection
15 April	**- Final implementation** **- Final report**
16-20 April	**Final Presentation**
2 May	**Project Exhibition**
30 May	**Project Competition** (For selected teams only)

Table 2: Schedule of the remaining tasks

5 Conclusion

The objective of the project is to build a piano tutorial application on HoloLens that gives piano learners a better learning experience with its 2 guiding features: falling notes and colored keys. The main difference of this application compared to existing solutions is that all holograms are shown directly on the physical piano keyboard, which makes the tutorials easier to follow. The project uses the HoloLens and OpenCV during the development process. There are two major challenges to be addressed in this project: virtual asset alignment and design for effectiveness. The first objective of the project has been achieved with satisfactory results. The application is able to recognize the piano keyboard and align a hologram to it with good accuracy. The user is also allowed to manually align the holograms to achieve better accuracy. In the next phase of the project, the UI and the tutorial contents of the application will be implemented. Toward the end of the project, there will be a thorough quality assurance test involving selected participants from outside the development team to evaluate the quality of the application.

There is still a problem concerning whether the design of Pianow is actually effective in helping piano learners improve their skills, which requires in-depth research in later stages after the application is ready for basic use cases. More functionality such as virtual music sheet and automated tutorial creation should be added to make the application more usable. In the long term, the application is anticipated to change the way self-taught pianists learn and be a good foundation for future projects which aspire to exploit the potential of Mixed Reality in musical education.

References

[1] L. Pianos. (2015). Why are there black and white keys on the piano? [Online]. Available: https://livingpianos.com/uncategorized/why-are-there-black-and-white-keys-on-the-piano/.

[2] M. virtual piano teacher. (2013). Piano practice tips: The 12 most common piano practice mistakes and how to correct them, [Online]. Available: http://www.musiah.com/blog/2013/august/12-common-piano-practice-mistakes.aspx.

[3] H. F. Revolution. (2015). Top 10 health benefits of playing the piano, [Online]. Available: http://www.healthfitnessrevolution.com/top-10-health-benefits-playing-piano/.

[4] Thumbtack. (2017). How much do piano lessons cost? [Online]. Available: https://www.thumbtack.com/p/piano-lessons-cost.

[5] CMUSE. (2014). How to get your child excited to play a musical instrument, [Online]. Available: https://www.cmuse.org/how-to-get-your-child-excited-to-play-a-musical-instrument/.

[6] M. D. Dickey, "Teaching in 3d: Pedagogical affordances and constraints of 3d virtual worlds for synchronous distance learning," *Distance Education*, vol. 24, no. 1, pp. 105–121, 2013.

[7] C. Dede, "The evolution of constructivist learning environments: Immersion in distributed, virtual worlds," *Educational Technology*, vol. 35, no. 5, pp. 46–52, 1995.

[8] N. Piegdon. (2017). Synthesia, [Online]. Available: http://www.synthesiagame.com/.

[9] RealityDome. (2017). Exclusive interview with ayato fujii creator of teomirn app, [Online]. Available: http://www.realitydome.com/interview-ayato-fujii-teomirn-app/.

[10] S. Glickman, H. F.Y., S. Das, and B. Lee. (2016). Music everywhere, [Online]. Available: https://www.etc.cmu.edu/projects/music-everywhere/.

[11] Google. (2018). Synthesia, [Online]. Available: https://play.google.com/store/apps/details?id=com.synthesia.synthesia&hl=en.

[12] Apple. (2018). Synthesia, [Online]. Available: https://itunes.apple.com/us/app/synthesia/id579518445?mt=8.

[13] G. Comeau, "Playing by ear in the suzuki method: Supporting evidence and concerns in the context of piano playing," *The Canadian Music Teacher*, vol. 62, no. 3, p. 42, 2012.

[14] T. J. McCue. (2013). Why infographics rule, [Online]. Available: https://www.forbes.com/sites/tjmccue/2013/01/08/what-is-an-infographic-and-ways-to-make-it-go-viral/#432c05dd7272.

[15] F. Huang, Y. Zhou, Y. Yu, Z. Wang, and S. Du, "Piano ar: A markerless augmented reality based piano teaching system," *Third International Conference on Intelligent Human-Machine Systems and Cybernetics*, vol. 2, pp. 47–52, 2011.