COMP 4801 Final Year Project [2023/24]

Detailed Project Plan

Exploring the application of machine learning models to achieve advanced character control and improve the naturalness and immersion of virtual characters in games

Yu Ching Lok

Supervised by Dr. Choi, Yi King

Content Page

1 Project Background

As artificial intelligence technology continues to advance, it is being used in a variety of fields, including creating human-like conversations through ChatGPT and generating new images through Mid-Journey. While AI technology is primarily used in academia and business, it is also being used in entertainment. Virtual YouTubers are a great example of AI applied to entertainment, where the use of landmark detection technology allows users to control animated characters by capturing their facial expressions and body movements through the camera (Singh, 2023).

However, there are limitations to current methods of controlling avatars using AI. Due to the limitations of landmark detection models, they can only control basic body movements and simple facial expressions. Expensive motion capture systems are required to achieve more advanced control of character movements (Gank Content Team, 2023). In addition, changing an avatar's emotional expressions currently requires manual input (Gank Content Team, 2023), resulting in somewhat stiff expressions. These limitations hinder the interaction and naturalness between the user and the virtual character.

To overcome these challenges, I plan to explore the integration of different AI models into the control of virtual characters, rather than simply applying landmark detection models, and in particular adding emotion detection and gesture recognition models. By combining these models, we can provide a more unique and natural experience when controlling virtual characters. Importantly, this approach aims to keep the system economical by using only one camera, thus providing a better user experience without a significant increase in cost.

2 Project Objective

The main objective of this project is to explore how artificial intelligence models such as emotion recognition and gesture recognition can be used to enhance control over avatars in a more natural and unique way. In this project, I will create an application that will allow users to have advanced control over avatars, making their interactions feel more realistic. The application should provide following features:

1. Control the avatar's body movements.
2. Control the facial expressions of the avatar.
3. Express different emotions through the avatar.
4. Make the avatar perform special actions.

To do this, the application will use models to detect poses and facial landmarks for basic body movement and facial expression control. It will also use emotion recognition models to control the expression of emotions and gesture recognition models to trigger specific actions by the avatar.

3 Project Methodology

3.1 Application

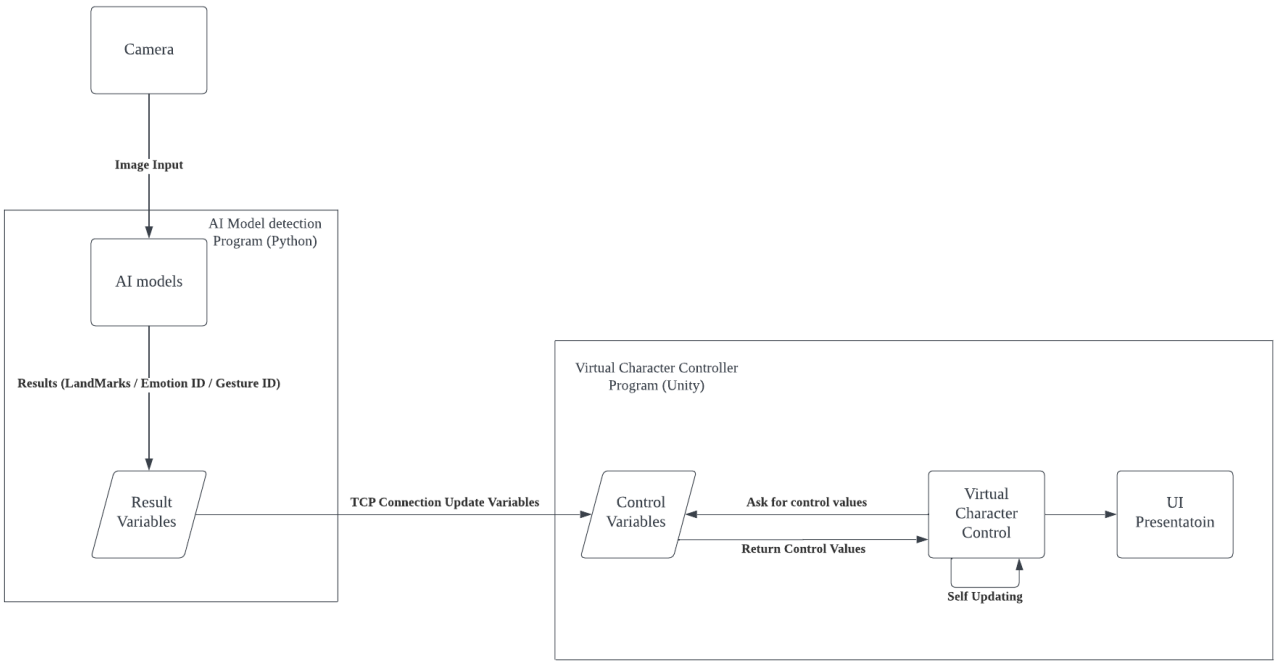

Figure 1: General Structure of application

The application will consist of two programs, one for AI model detection and the other for virtual character control. the AI model detection program will be written in python and the virtual character control program will be written in C#, using unity as the platform. The details are described in following sections.

3.2 AI Model Detection Program

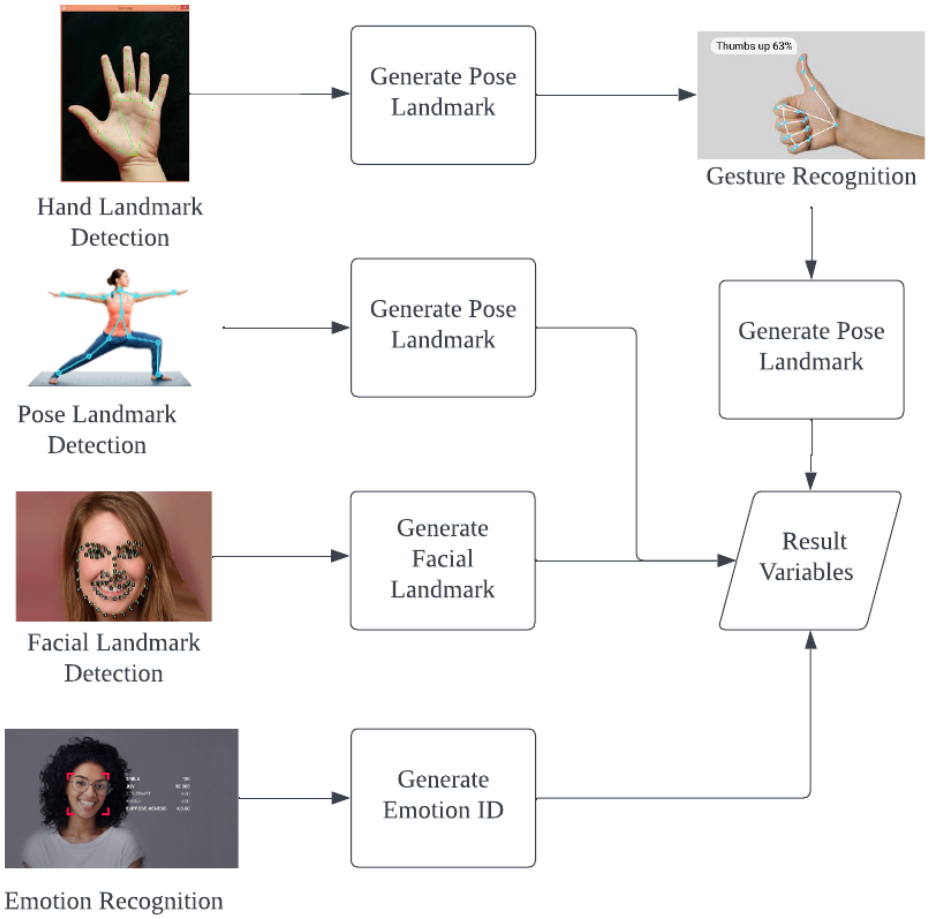

Figure 2: General structure of AI model detection program

This will be a program written in python that will only be responsible for tasks related to AI model detection. Once the program receives the image stream from the camera, it will feed it into the AI model for detection and the detection results of different AI models should be stored in the appropriate variables. The result variables should be sent to the unity program periodically over TCP so that the virtual character can be updated based on the detection results of the AI models.

3.21 Pose Landmark Detection and Facial Landmark Detection

We have chosen the Mediapipe holistic model as the primary AI model for detecting postural and facial landmarks. The model provides comprehensive support for recognising landmarks associated with faces, poses and hands. Notably, it shows impressive real-time performance even on medium-spec devices and web browsers. For example, it achieves a frame rate of 20 FPS on devices such as the Samsung S9+ (Grishchenko & Bazarevsky, 2020).

We decided to use this model mainly because its lightweight architecture allows for efficient execution on a wide range of devices. In addition, the model consistently provides accurate and reliable landmark detection results. By utilising the functionality of the Mediapipe holistic model, we can ensure efficient and reliable landmark detection, thus improving the overall speed and performance of the program.

3.22 Emotion Recognition

Regarding emotion recognition, the specific model details are yet to be confirmed. However, we have decided to develop a custom model to achieve the most suitable results. This approach allows us to have greater control over the model and obtain the desired performance. Currently, we are conducting experiments with different AI models to determine the most suitable one. Generally, our idea is to either use the facial landmarks generated by the Mediapipe holistic model as input for another classification model to perform emotion classification or directly use the raw image as input for emotion classification. We are currently testing an input classification model based on facial landmarks using the MLP architecture. However, the performance of the model is not satisfactory, so we are actively exploring ways to improve its effectiveness. In addition, we are working on other classification models, such as using landmarks for action unit recognition and then using the results for emotion recognition, and using a CNN-based AI model for emotion classification using raw images as input. Our current research focuses on exploring various AI models to develop classification models specifically for emotion recognition. At this stage, the research is still in the exploratory phase and many important details are yet to be confirmed. We are conducting extensive testing to determine the most appropriate model for our purposes. Once the experiments are completed and a final decision is made, we will discuss and share more details of the selected models.

For emotion recognition, we have decided to use the FER-2013 (Facial Expression Recognition 2013 Dataset) dataset as our foundation dataset. The FER-2013 dataset consists of approximately 30,000 facial RGB images representing seven different emotion classes. The emotions include Angry, Disgust, Fear, Happy, Sad, Surprise, and Neutral. (PapersWithCode) Our initial models will be built based on this dataset, hoping to classify between the seven emotions. We will conduct testing to test the performance of the AI models using evaluation metrics such as F1 score, accuracy, precision, and recall. It is crucial to ensure that the models deliver a relatively good performance on this dataset.

Once we have established a satisfactory performance on the fer2013 dataset, we will proceed to test the models on a custom dataset. We plan to create a customized dataset by capturing the data ourselves to closely simulate real-world application environments. This approach will allow us to evaluate the performance of the model in realistic scenarios and ensure its effectiveness in real-world environments. Additionally, this custom dataset will allow us to fine-tune the model for optimal performance in real-world environments if the model's generalization capabilities do not meet our expectations.

And by testing and fine-tuning the models on fer2013 dataset and custom dataset, we aim to develop a reliable gesture recognition models that can effectively perform in our applications.

3.23 Gesture Recognition

Similar to emotion recognition, our approach to gesture recognition involves creating a custom model and evaluating and fine-tune it using both public and custom datasets. We intend to use hand landmarks from the Mediapipe holistic model as inputs for a classification model, enabling hand gesture recognition. The primary training and evaluation dataset for our study will be the HAGRID dataset (HAnd Gesture Recognition Image Dataset). It consists of 552,992 RGB images categorized into 18 gesture classes. The dataset encompasses diverse environments, including scenarios where subjects face or back a window. The hand gestures were performed at distances ranging from 0.5 to 4 meters from the camera (Kapitanov, 2022). We have recently developed a simple multilayer perceptron (MLP) neural network for classification purposes. Initial experiments on the HAGRID dataset achieved promising results. In addition, the model consistently demonstrated efficient performance when tested in real-time using a camera stream capturing myself performing hand gestures, suggesting a satisfactory generalization capability. However, more testing is needed to determine if any adjustments or improvements to the model are necessary.

In summary, our future research will require more experimentation and exploration using different AI models to evaluate their performance and determine the most suitable model for our intended purpose. Subsequently, we will develop an application that integrates these AI models into a unified detection program dedicated to our target application. This iterative process of experimentation and refinement will allow us to improve the performance and functionality of our AI models to ensure that they effectively meet the requirements of our applications. While the previous chapters have mainly emphasized the development of AI models with limited discussion of their application in our project, we will delve into the application of AI models to the control of avatars in the following sections

3.3 Virtual Character Control Program

This part will be a unity program written in C#, responsible only for the control and display of virtual characters. It will contain a TCP server that accepts the python program as a client and continuously listens for updates on its detection results. The received detection results will change the control values of the virtual characters accordingly. While listening for control values, the virtual character shall be constantly updated to update its state and determine how the avatar is displayed.

In order to facilitate the virtual character control procedure, we decided to implement 3D virtual character control. Our main focus is to utilise the "Unity Chan" model as our main avatar. We will design a specialised program to control the overall movement of the "Unity Chan" avatar. To this end, we will conduct an in-depth exploration of the control mechanisms associated with the "Unity Chan" 3D avatar. Through experimentation and analysis, we will develop project-specific control procedures. More details about the control mechanisms will be discussed once our exploration and experiments are complete.

As mentioned above, the application shall allow the user to control the virtual character's body movements, facial expressions, emotional expressions and special actions. The following are the details of how each of these features will be implemented.

3.31 Body Movement Control

In order to control the body movements of the virtual character, we will utilise a pose landmark detection model. The model will analyse the input images from the camera stream and generate pose landmarks representing the position of various body parts. By updating the body parts of the virtual character based on these landmarks, we will be able to achieve realistic body motion control.

3.32 Facial Expression Control

To modulate facial expressions, we will use a facial landmark detection model. Similar to body movement control, this model will analyse facial features in the input image and determine the location of specific facial landmarks. By using these landmarks, we can manipulate the facial expressions of the virtual characters to achieve realistic and expressive interactions.

3.33 Emotion Expression Control

In order to express emotions through virtual characters, we will utilise an emotion detection model. The model will analyse the input images from the camera and identify the emotions expressed by the user. It will assign an emotion ID to each frame so that the virtual character can show the corresponding emotional expression, thus enhancing the overall realism and engagement of the interaction.

3.34 Special Action Triggering

In order to trigger special actions of virtual characters, we will use a hand gesture recognition model. The model will analyse the hand landmarks obtained through the hand landmark detection model and identify the gestures that the user intends to make. Once a gesture is detected, a signal will be sent to the virtual character control programme to activate the corresponding special action of the virtual character, thus enhancing the interaction experience.

In summary, the AI model will play a key role in controlling the avatar in all aspects including normal postural movements, facial expressions and special gestures. The virtual character control program will continuously monitor and receive updates from the AI model detection program. These updates will provide control values that determine the behavior and actions of the virtual character, ensuring that it accurately reflects the inputs and commands of the AI model.

4 Project Schedule and Milestones

Regarding the project schedule, I have outlined a general plan. Initially, I will focus on exploring and experimenting with various AI models. Once I have evaluated these models, I will proceed to develop the virtual character control program. Subsequently, I will consolidate all the components and create an application. If there is any remaining time, I will dedicate it to conducting further research on improving the character control program and AI models, or exploring the development of additional features. The detailed schedule is as follows:

Date	**Task**	**Status**
Early September 2023	- Search and exploration of AI models, Specifically Models for emotion recognition and gesture recognition - Set up GitHub repository	Finished
1, October, 2023	- Complete Detail Project Plan - Set up Project Web Page	Finished
Mid October, 2023	- Evaluate performance of AI models - Finalize suitable AI models for development - Pipelining of AI models to evaluate performance	Pending
Mid November, 2023	- Development of virtual character control program - Linkage of Ai model detection program and virtual character control program	Pending
8-12 January, 2024	- First presentation	Pending
21 January, 2024	- Complete detail interim report - Complete preliminary implementation	Pending
Mid February, 2024	- Further improvement of AI models and virtual character control program	Pending

Mid March, 2024	- Finalize development of AI models and virtual character control program	pending
15 – 19 April, 2024	- Final presentation	Pending
23 April, 2024	- Complete Final report - Complete application	Pending
26 April, 2024	- Project exhibition	Pending

Reference List

1. Kapitanov, A. (2022). *Hukenovs/hagrid: Hand gesture recognition image dataset*. GitHub. https://github.com/hukenovs/hagrid
2. PapersWithCode. (n.d.). *Papers with code - fer2013 dataset*. FER2013 Dataset | Papers With Code. https://paperswithcode.com/dataset/fer2013
3. Grishchenko , I., & Bazarevsky, V. (2020, December 10). *MediaPipe holistic - simultaneous face, hand and pose prediction, on device*. MediaPipe Holistic — Simultaneous Face, Hand and Pose Prediction, on Device. https://blog.research.google/2020/12/mediapipe-holistic-simultaneous-face.html
4. Singh, J. (2023, May 21). *What is a vtuber, and how do you become one?*. Cointelegraph. https://cointelegraph.com/news/what-is-a-v-tuber
5. Gank Content Team. (2023, July 8). *What is a vtuber? The Ultimate Guide to Virtual YouTubers!*. Gank Blog. https://ganknow.com/blog/vtuber/