MASTER OF TECHNOLOGY PROJECT REPORT

Gesture Controller Game System

Luo Tianchen, Kuang Shan, Xia Xu, Tong Shiqing

30th October 2022

Contents

1 Executive Summary

As one of the industries expanded by the advancement of computer technologies such as computer vision and machine learning in the 21st century, the game industry has huge commercial value and market. Taking Shanghai, China as an example, according to the Shanghai Game Publishing Industry Report, the revenue of the game industry in Shanghai was RMB 125.03 billion, a year-on-year increase of 3.6%. The related industries driven by the game industry also have a broad market, such as e-sports events. However, according to the data in the "Report", the number of game users in China will be about 667 million in 2021, and the growth rate will hit a new low in recent years. This shows that under the current demographic background, the number of consumers that the game market can absorb is close to the upper limit. How to attract more new players or attract players who have played traditional games to try new games based on the current number of players is a business problem worth solving and thinking about.

As a game, its goal is to give different feedback and continue to attract players' attention through effective interaction with players, motivating players to continue playing. In a game system, many features can be developed to attract the player's attention. The most common way is by giving players a certain number of rewards. However, this method also has limitations. After the player receives a certain number of rewards, the player will be tired of the rewards given by the game, such as medals and special items. Players will be less interested in the game.

To solve this problem, our team decided to slow down the decay rate of players' interest by modifying the interaction between the system and players based on the perspective of the game system and player interaction. Traditional games accept the user's keyboard/handle or mouse as input to interact with the user. In our system, we will analyze the user's intention by collecting the user's picture information, so that the user can play different games without the need for external input devices such as a keyboard, mouse or gamepad.

In this system, we utilize the techniques learned in the ISS course by integrating a real-time user hand information collection module, a user hand feature extraction module, a user intent analysis module, a user background replacement module, and three independent interactive game modules - Snake, rocket dodging obstacles and quiz games and user rights management modules to develop intelligent game systems.

At the same time, with the improvement of people's consumption level, more and more enterprises attach importance to brand management. Including, but not limited to, almost 15% annual increase in papers at HCI conferences using interactive techniques to improve users' perception of corporate branding. Based on this, we propose our system's application scenarios and business value. In popular restaurants or shopping malls, customers' children will be dissatisfied due to long waiting (such as waiting for customers to pay remaining for customers to shop); we hope to provide our game system allows customers' children to be quiet and patient during the waiting period, thereby improving customer

satisfaction and ultimately improving the corporate brand image.

The role of interaction is far more significant than our imagination. We believe that our system has great potential in the future. It may be able to provide new ideas for game companies to promote games. Imagine the following scenario: our software embeds more new games; it can be directly installed on various machines; then placed in shopping malls or other high-traffic places; users simply pass in front of it and have a direct opportunity to stop and play new games with simple finger interaction (avoid learning any game operation, allowing users to face the game itself directly); we believe that such a new type of game advertising has excellent value.

2 Market Research

In this chapter, we conduct business analysis on our products, including the following aspects: 1. Business model and solutions 3. Market sales 4. Competitive product analysis.

2.1 Business Model and Solution

The target sales of our system are corporate partners such as catering or subways. And our application scenario is in a popular store or restaurant, the customer's child will be dissatisfied because of the long waiting time for the customer to pay or complete other things. Based on this, We developed a gesture-based game system so that children can play while waiting. For businesses, our software can enhance the brand image of the business. For customers, our software can prevent customers from handing their mobile phones to children while waiting, and then the children send wrong messages or cause damage to the mobile phone system.

Our services can be divided into software as a services (SAAS) and software customization. First, we provide our software for users to install and use for SAAS. The functions of the software include 1. Users register and log in to the system. 2. Users use gesture control to play three games for different age groups. Secondly, we provide the following three improvements for software modification: 1. Add games; we can develop and add different games to cover more age groups according to their needs. 2. Add advertisements; we can add toddler advertisements to promote the products of corporate customers in the games we develop. 3. Interface addition; we can add interfaces to our software for different games.

2.2 Marketing and Sales

Market our products and services directly to potential corporate clients through direct sales and long-term partnerships. We attract and retain customers by providing outstanding integrated technology solutions, professional services, and excellent customer service. We are responsible for providing ongoing support services to our customers. Our developers participate in trade fairs, speak at panel discussions, publish articles and papers, etc., to enhance the sales of our products.

2.3 Analysis of Other Products

Analysis are divided into three categories: 1. Products with Similar function, 2. Other products with gesture control. For each category, we use SWOT for analysis.

2.3.1 Products with Similar Function

The products with the same type of functions as our software is Arcade Cabinet. It is used by coin-operated, and the main games are basketball shooting, airplane wars, etc. The four dimensions of our products are compared as follows:

Strength: 1.Our products cost less than game consoles and are easy to carry and move (only a machine with operating system). 2. Our products use gesture control. Users do not need to learn interaction methods, simplifying the process.

Weakness: Our products are inferior some cabinet in terms of sound and picture.

Opportunity: The rise of virtual reality technology has led people to like intelligent interaction methods such as gesture interaction.

Threat: Our games may not be compatible with specific platforms.

2.3.2 Other Products with Gesture Control

Other products of gesture control are currently virtual technology products such as AR. The basic implementation method of AR is to generate a realistic virtual world with the help of computer technology and other equipment. The four dimensions of our products are compared as follows:

Strength: 1.Our products occupy fewer resources than other virtual technology products. 2.Solve the user's problem in the application scenario.

Weakness: Our products are only for children.

Opportunity: Currently, the scene where users wait is becoming increasingly familiar with the improvement of consumption level.

Threat: Some businesses may be willing to spend a lot of money on equipment to deploy AR products for customers to use.

3 System Architecture and Description

3.1 System Architecture

The Figure 1below displays the architecture of the system.

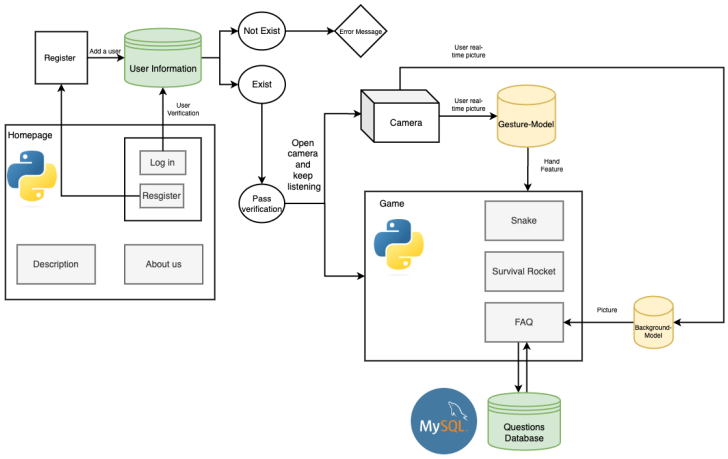

Figure 1: System Architecture

Frontend(Python):

1. Handle the interaction with the user.
2. Display system different pages including the homepage, about us page, description page, game page and three different game interfaces.

Backend(Python):

1. Connect to the database and query results.
2. Check the access permission of the users.
3. Launch different games based on the request.

Database(MySQL):

1. Store the questions id, questions, and answers as a table of game FAQs
2. Support query and return target results to the game.

Game(Pygame):

1. Provide game modules for the system.

Model: A different model is used in this project to handle different requests and return results.

1. Hand model: a media model that is used to analyse and extract users' gesture features to return 21 3D landmarks from a single frame.

2. Gesture model: Calculate and map the 21 3D gesture landmarks to four directions

3. Selfie- segmentation model : a media model that is used to split the prominent humans from the scenes.

3.2 System Description

The figures below display the two main data flow of the system and how the system use and processes the data. Blue boxes represent the module-related data. The first one is the user management data flow. The user data is passed and checked on the Homepage. As displayed in the figure, the user who visits this system is required to log in first. The login module will access the user data to verify whether the user has permission to use this system. Once the permission is checked, the user with a valid username and password will be redirected to the game page. The unauthorized user will be refused to use this system. They will be informed to register them first and store valid user information. The second figure indicates the most significant data flow of the game. For this system, user image information is collected as the main data by the camera. Considering the balancing of the performance of the game and the accuracy of analyzing the user's intention, the camera is working at 15 FPS. The collected pictures will be proceeds in two different ways. For the game Snake and Rocket, the pictures will be analyzed by the model and extracted 21 main features including wrist features, thumb features and other four fingers' features. For the next step, the gesture features will be used for different games. For the rocket, we take the INDEX_FINGER_TIP as the main input data to present the rocket position. However, for the game snake, the gestures feature needs to be further processed. We use a model called gesture model to map the gestures feature to the four directions. And then, the direction data will be used in the game Snake. For the FAQs game, the intake pictures will be proceeded by the model Selfie- segmentation model to recognize and split the player object from the background and generate a new background. MySQL is used to import some default data of questions and answers to the game.

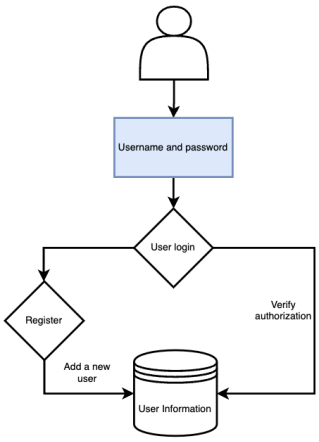

Figure 2: Dataflow of user part

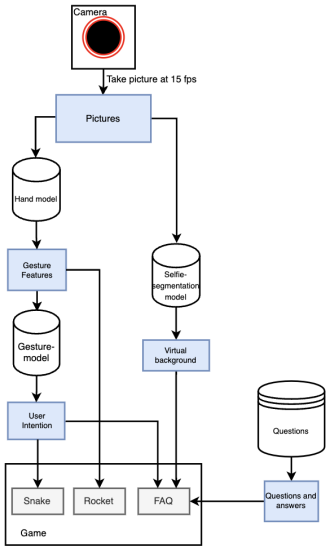

Figure 3: Dataflow of game part

4 System Implementation

4.1 GUI

4.1.1 The introduction of Tkinter

We use this package for our GUI interface. It is a Python interface wrapper around the TCL/TK toolkits and the standard library that comes with Python and supports cross-platform operation. Among the add-on modules for the Tk interface, Tkinter includes several Python modules called tkinter. There are two important modules, one is Tkinter itself, and the other is called Tkconstants: the former automatically imports the latter. Essential functions are shown in Table 1.

Table 1: TKinter function

Notations	Description
root=Tk()	Generate main window
root.geometry('250x250')	Change form size ('width x height')
root.title('name')	Change the position of the form ('+ abscissa + ordinate')
root.mainloop()	Show main window
root.winfo$_s creenwidth$()	Get screen width
root.winfo$_s creenheight$()	Get screen height
root.iconbitmap('favicon.ico')	Settings window icon
root.["background"] ="7AC5CD"	Set background
root.attributes("-alpha", 0.6)	Set window transparency

4.1.2 The coding process of our GUI

We first made a welcome page that includes the function for registering and login, user can create their own account in our system or use admin permission to use our system. (Figure4)

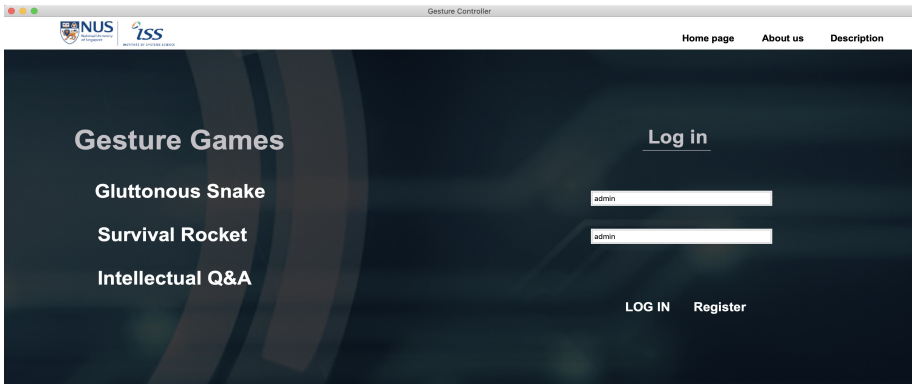

Figure 4: GUI Welcome page

The main coding structure is shown in Figure 5, we use tkinter to simultaneously create 5 different pages (python files):

1. GUI-Main: Welcome page that provides the function for login and user registering.
2. GUI-instruction: Give a brief introduction and description of our intelligent system.
3. GUI-AboutUs: Our teammates' information and main individual contribution.
4. GUI-Register: Provide the register function and link will backend user-info storage. It will check users' password and record username locally.
5. GUI-games: The page for game selection, it will invoke python subprocess to open other pygames, gesture controller, and classification model altogether.

Based on tkinter functions, we made our own button, bar, and icon style, integrate sub-process, page turns and SQL functions into the 5 main GUI codes.

```
class IntelligentSystem(tk.Tk):

    def __init__(self, *args, **kwargs):
        tk.Tk.__init__(self, *args, **kwargs)

        container = tk.Frame(self, width=500, height=250)
        container.pack(side="top", fill="both", expand=True)
        container.grid_rowconfigure(0, weight=1)
        container.grid_columnconfigure(0, weight=1)
        container.grid_propagate(0)
        container.pack_propagate(0)
        self.frames = {}


        for F in (Welcome_Page, us.About_Us, ins.Instruction, game.Games, register.Register):
            page_name = F.__name__
            frame = F(parent=container, controller=self)
            self.frames[page_name] = frame
            frame.grid_propagate(0)
            frame.grid(row=0, column=0, sticky="nsew")


        self.show_frame("Welcome_Page")

    def show_frame(self, page_name):
        '''Show a frame for the given page name'''
        frame = self.frames[page_name]
        frame.tkraise()
```

Figure 5: GUI structure code

4.2 Database

4.2.1 The introduction of database

As is known to all, data in files are hard to search and modify, and the security of the files can not be guaranteed. Due to these reasons, database becomes a more safer and convenient method to store data. One of the most popular database which is also used in our project is MySQL. It is implemented in a "client/server" mode and is a multi-user, multi-threaded small database server. What's more, MySQL is open-sourced, which means anyone can get the source code of the database and fix the bugs of MySQL. MySQL has cross-platform

features, it can be used not only on Windows platform, but also on UNIX, Linux and MacOS platforms. Compared with other databases, MySQL is more convenient and faster to use, and MySQL is free and has low operating costs. Therefore, more and more companies begin to use MySQL.

4.2.2 The Use of MySQL

After installing the MySQL server successfully in the computer, you can now have access to the database. You can easily create databases and tables in terminal. Also you can add, delete, search for rows, columns in a fast way. Moreover, there is also some browsers such as navicat for MySQL that can make the data modification much easier.

4.3 Mediapipe

As to detect the hand movements presented in the video, we use MediaPipe. MediaPipe is a multimedia machine learning model application framework developed and open-sourced by Google Research. At Google, a number of important products, such as Google Lens, AR Core, Google Home, and so on have deeply integrated MediaPipe.

4.3.1 Mediapipe function

MediaPipe offers cross-platform, customizable ML solutions for live and streaming media. In general, the Mediapipe have many functions, such as Hands, Pose, Face Mesh and so on. In this project, we mainly use the Hands to get landmarks and Selfie Segmentation to remove the backgrounds.

4.3.2 Mediapipe Hands

In the process, each hand is represented by 21 landmarks which contain x, y, and z. Among them, z represents the depth of the landmark, proportional to the distance to the camera. During processing, multiple machine learning models work together. A palm detection model operates on the full image and returns an oriented hand bounding box, which contains the whole hands. Compared with face detection, hands don't have reliably visual features like eyes and mouths. So in the mediapipe, they train a detector to detect palms instead of hands, since estimating bounding boxes of rigid objectives is more simpler. Then they use an encoder-decoder feature extractor for bigger scene context awareness. Lastly, minimize the focal loss and get a good result. After detecting the palms, a hand landmark model operates on the cropped image region defined by the palm detector and returns high-fidelity 3D hand keypoints, just as Figure 6 shows.

When the process function inherited from Solutionbase is called, the results will contain MULTI_HAND_LANDMARKS, MULTI_HAND_WORLD_LANDMARKS and MULTI_HANDEDNESS. MULTI_HANDEDNESS returns the label and value. The label conveys which hand is detected. And the score represents

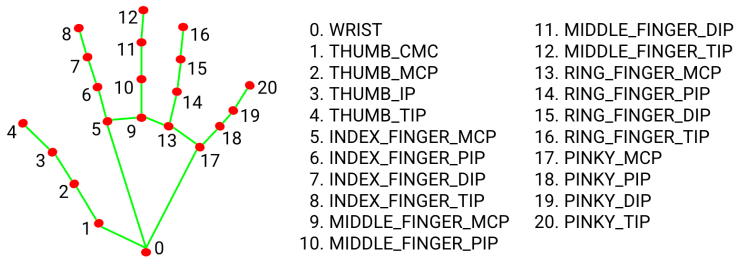

Figure 6: The 21 landmarks

the score of the detected hand. The relatively most significant parameter is the MULTI_HAND_LANDMARKS. It stores the positions of the 21 landmarks. Just as showing in the Figure 7

Figure 7: The 21 landmarks

With these 3D landmarks, more work can be done. For example, it can check if the finger is up or not, by comparing the values of y or x. The first function used in this project is checkFingerPositionABCD. We divide the full screen into four areas, representing A, B, C, D respectively, which can be inferred from the function name. We call the checkFingerUp function first and return none if there is no finger up. The finger we mainly focus on is the index finger, which may make a pointed action. We only need to locate the index finger's tip and we can get the position. This function will return actions detected as 'A', 'B', 'C' or 'D'. So that the results can be used in the region detected and FAQs games which will be introduced later. Besides, we can finish more complicated

work by these landmarks. Once we get all the positions of the landmarks, we can calculate the distances and angles between each finger. Using the distances and angles, we can predict what gesture is being presented. This classification model will be illustrated later.

4.3.3 Mediapipe Selfie Segmentation

The selfiesegmentatiton module in the Mediapipe's solutions offers two kinds of model: general or landscape. Both models are based on the MobileNetV3 network, with modifications to make them more efficient. When this function is called, the general model operates on the tensor and outputs a 1D tensor representing the segmentation mask.

To implement the background removal function, we use the removeBG function from CVzone. The CVzone is a package containing many convenience functions and makes it easier to detect faces, track hands, image process and so on. And this package mainly uses Mediapipe and OpenCV. The module we are going to use is called SelfieSegmentationModule. In this module, the input image will be resized automatically to the desired tensor dimention and fed into the pre-trained model. The removeBG will return a 1D array representing the segmentation mask. Then the numpy's function stack is used to expand the mask array to three channels. When calling the function, another parameter threshold is also to be set. If the pixel of the stacked array do not exceed the threshold, then this pixel will be replaced by white background or given picture. As a result, the larger the threshold is, the more pixels will be cut. The differences between different thresholds are showing in figure 8.

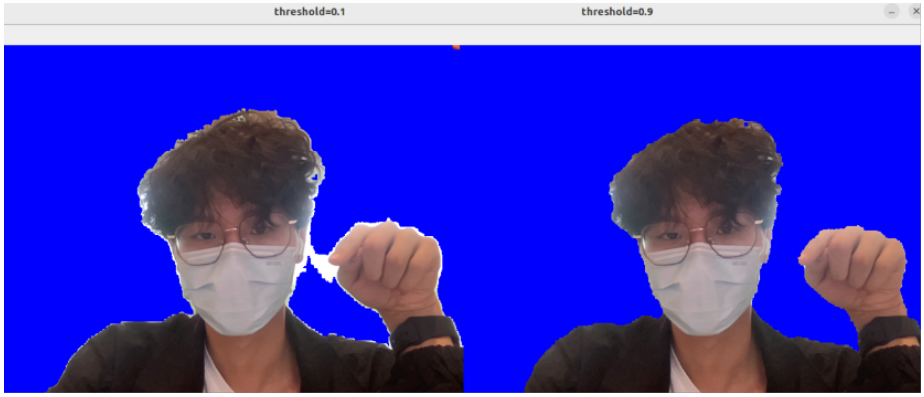

Figure 8: The differences between different thresholds

4.4 Gesture Classification Model

By invoking and using the device's camera, our system is able to capture every real-time frame and process it as RGB images. Firstly, we use the Mediapipe model from Google Research to localize human hands and its 21 hand joints' 3D coordinates(see figure 6). Since these 21 coordinates represent the whole hand, when users are moving hands or changing their hand gestures, these coordinates will change correspondingly. Based on these observations, we made a hypothesis: These 21 hand joint coordinates can be considered useful features of hand gestures, and we can predict the hand gestures by analyzing these features.

4.4.1 Feature Collection

In order to explore the potential expressive ability of hand features, we collected thousands of features of hands by video-processing:

Figure 9: Feature collection examples

1. Using open CV to temporarily save every frame as RGB image captured by laptop camera.
2. Using Mediapipe Model to process each frame and return the coordinates of hand joints coordinates.
3. For every joint, subtract the coordinate of WRIST, in this way, we gain the relative position of each joint without considering where the user's hand is located on the screen in order to remove useless spatial information and improve the later classification model accuracy.
4. Record the pre-processed features above in CSV format file, the hand detected in each frame corresponds to a single row of training samples.

5. We invited 4 people to record videos by posing four different hand gestures in front of the camera (see figure 9), we plan to train a model that is able to correctly classify these four gestures and control the Gluttonous Snake game, e.g. thumb up will let the snake move upward.

After finishing gestures features collections, some experiments are made for selecting the most appropriate Machine Learning model by implementing 10-fold cross-validation for model comparison.

4.4.2 Gestures Classification Model

Our group has 4 members, each of us recorded a video for four gestures, in the end, we prepared 10k+ gestures features for model training. Since we are looking for the most appropriate model that is able to balance the trade-off between accuracy and inference speed, we choose some candidates: SVM, MLP (two hidden layers), Logistic Regression, Bayes, and Decision Tree. The reason that we didn't choose Deep Learning models is their lower inference speed, we require the model can process each gesture as much quicker as possible in order to increase the GUI response speed and improve user experience.

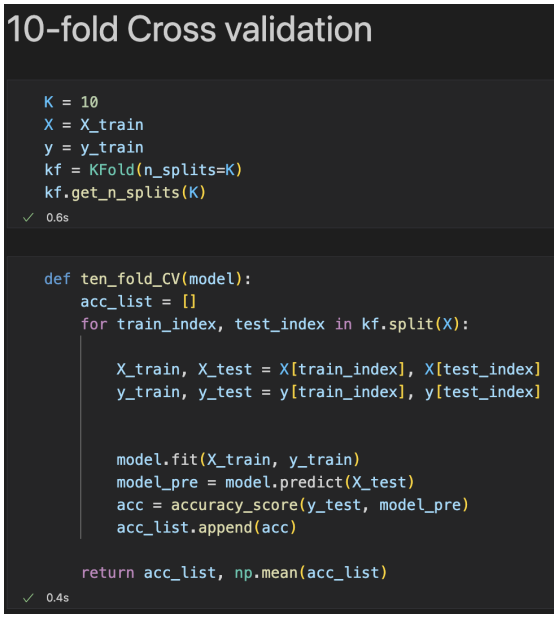

Figure 10: 10-fold Cross Validation Code

For these Machine Learning candidates, in order to find out which one fit our object best, a ten-fold cross-validation experiment is conducted. By randomly splitting the self-collected hand gestures features' dataset into 10 folds, each time we choose a single fold of data as the testing set and 9 fold data as the training

set to do cross-validation for the sake of selecting the Machine Learning model with the best performance. Here we list the average accuracy of each model among all ten fold cross validation stages:

Model	SVM	MLP	Logistic	Bayes	Decision Tree
Accuracy	0.993	0.998	0.991	0.987	0.989

Since cross-validation gives information to assess how well a classifier generalizes, from the result of experiments, we see that MLP has the best performance, on the other hand, it has a similar inference speed compared to other candidates. We choose it as the gestures classification model in this project and use its result to control the Snake moving direction.

4.5 Model Deployment

After evaluating the model's expected performance and capability by 10-fold cross-validation, we chose to use MLP as our gesture classification model, responsible for correctly classifying the game players' commands.

In order to deploy our model and combine its result with our system, we use the 'pickle' package from python to save the weights of model into local file, in this way, we don't need to train the model every time before using, the model can be conveniently loaded and put into practice.

Besides models, we also create a JSON file stored locally for model result translator; since our classification model's output is category, we have to define the specific meaning of each output and broadcast it to other sub-process. For example, in our gluttonous snake controller system, the thumb-up gesture is the first category meaning go up, thumb point in the left direction is the second category let the snake turn left.

After looking up the actual meaning of the gesture from the JSON file, our system 'understand' what is user's intention, and execute the system to follow user's instruction(Figure 11). When designing our system, our main usage scenario is offline, and hope our system can fit every device with a camera(future work), therefore, the model should be deployed locally to avoid bad internet connections and transmission. We also left a convenient interface for people who are interested in our system to replace the gesture classification model on their own.

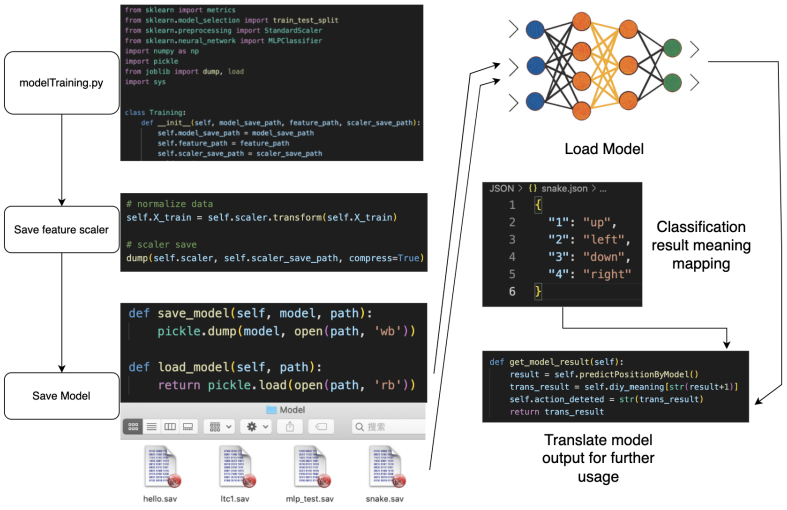

Figure 11: model deployment

4.6 Model Result Usage

Our projects main function is three interactive gestures control games give examples for gesture classification use cases and let user to use hand gestures for game playing instead of keyboard and mouse. Our MLP Model achieved 99.8 percentage accuracy in classify four hand gestures to control the moving direction of Gluttonous Snake. When playing the game, the camera of laptop is invoked and shows up a simple interface displaying the instant real-time video captured by the camera, user is able to see what their gesture classified by our model(Figure 12).

Meanwhile, the images captured by the camera will be processed by Mediapipe to recognize the human hand and provide the coordinates of 21 hand joints for each frame, we use these coordinate as features to classify what gesture it is.

Our our model inference speed constraint the hand gesture processing fps rate to around 20 frames per second (MacBook Pro), for each frame, we will reform the 21 joints coordinates into a vector. Then, normalize the vector by using the Normalizer Scaler we trained based on our training data. After normalizing the input vector, we pass the vector to our model for further classification, based on the model output, our code will lookup the meaning of user's gesture and control the game to follow user's commands. From Figure 12, we can see that when user pose thumb up gesture, the model can correctly classify it and display the current result in the windows, at the same time, the gluttonous snake in the game moved up. During our system developing process, we made many experiments on adjusting model using and its connection between input and output. Eventually, we achieved satisfied interactive game control performance, user can operate the game fluently by posing gestures in front of the camera.

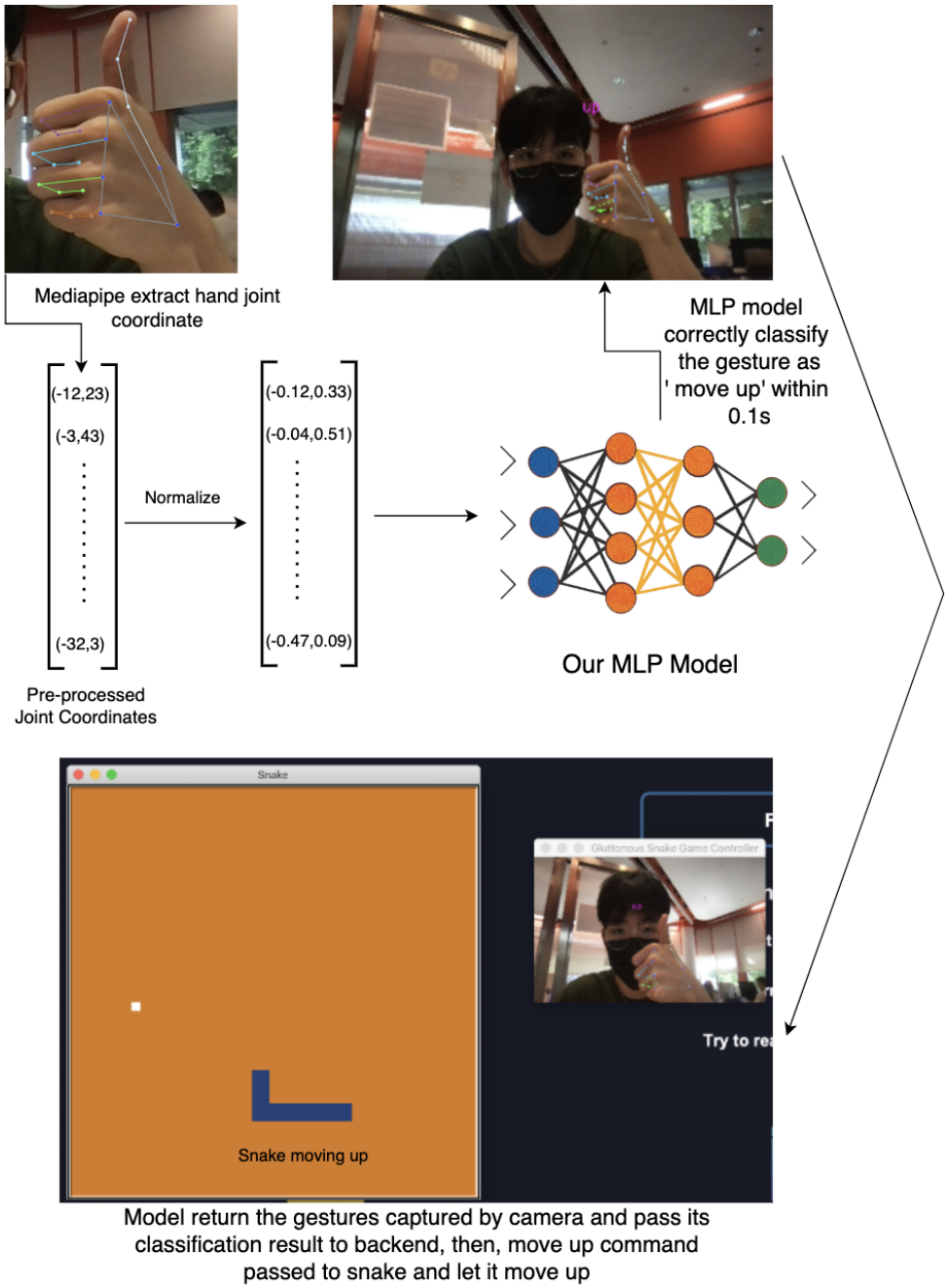

Figure 12: model usage

5 Appendix

5.1 Appendix A: Project Proposal

GRADUATE CERTIFICATE: Intelligent Reasoning Systems (IRS)

PRACTICE MODULE: Project Proposal

Date of proposal:

2 OCT 2022

Project Title:

ISS Project – Gesture Controller Game System

Sponsor/Client: *(Name, Address, Telephone No. and Contact Name)*

Institute of Systems Science (ISS) at 25 Heng Mui Keng Terrace, Singapore

NATIONAL UNIVERSITY OF SINGAPORE (NUS)

Contact: Mr. GU ZHAN / Lecturer & Consultant

Telephone No.: 65-6516 8021

Email: zhan.gu@nus.edu.sg

Background/Aims/Objectives:

The game industry has broad prospects and a huge commercial market. But common games usually interact with players via keyboard, mouse and gamepad. Players will inevitably feel tired and bored after continuous play.

Our team decided to design and develop a game system. This game system consists of multiple sub-games and has basic system-owned functions such as access control. The game system will change the mode of the traditional game and player interaction and adopt a more novel mode. Our game system will collect and analyze the user's image information in real-time to understand the player's intention and send the player's intention to the game to achieve the effect of accepting the player's input without using the mouse, keyboard, and handle. The game can be played if the device is equipped with a camera.

We want to appeal to two groups of players, the first is players who don't have a lot of gaming experience and who are interested in trying more different games to explore their favourite game types. The second is players who are already tired of the games they play regularly and are more likely to try out genres they haven't seen before.

Requirements Overview:

- Research ability

- Programming ability (Python)
- System design and developing ability

Resource Identified:

1. Dataset for model training

   10k+ self-collected hand gestures features, captured by camera and use mediapipe to extract hand joints coordinates, reduce noisy elements and normalize the coordinates, save as features for further model training.

   Questions and answers data for game FAQs
2. Hardware

   Camera
3. Software

   MySQL: FAQs

   Pygame: Snake Game, Rocket Survival Game

   MLP machine learning model: selected by implemented 10-fold cross-validation among other machine learning models, choose the most appropriate one for our project.

*Optional

Number of Learner Interns required: (Please specify their tasks if possible)

A team of four project members:

1. Luo Tianchen
2. Kuang Shan
3. Xia Xu
4. Tong Shiqing

5.2 Appendix B:Techniques and skills

Course Module	Techniques and Skills
Machine Reasoning	• Rule-based inference • SQL database • Intelligent system design •
Reasoning Systems	• Machine Learning models and cross-validation. • Deep Learning with neural network (Mediapipe pre-trained model, background removal) • Data pre-processing before model training.
Cognitive Systems	• Computer vision human interactive interface • Object detection • Feature extraction

5.3 Appendix C: Individual Report

Name: Luo Tianchen	**Matriculation Number: A0261747X**

Personal contribution to group project:

Following are my contributions in Gesture Controller Game System, IRS project:

1. Gesture feature collection and preprocessing:
   I made the code for gesture features collection and clean-up program: Firstly, using Mediapipe model to process each frame captured by camera and return the coordinates of 21 hand joints in X-Y axis. Preprocessed the coordinates and save as features for hand gesture, record in CSV format for further model training.
2. Gesture classification model training and selection:
   To assess how well a Machine Learning model generalizes, make 10-fold cross-validation to balance the tradeoff between accuracy and inference speed, compare the performance of among SVM, Logistic Regression, Bayes, Decision Tree, and MLP. Choose MLP as our model for gesture classification and trained the final version model for actual system usage.
3. Model deployment and usage:
   Finished training, encapsulation, and loading of the Machine Learning model in our system, combine the model output with the system and GUI code. Testing and optimizing our system's user interactive experience. Work together with teammates on system design and GUI code implementation on tkinter.
4. Gluttonous snake game's gesture controller
   By invoking camera of laptop and process the image captured, extract and process gesture features instantly and using ML model to inference the category, by interpreting the output of our classification model, control the snake game in multi-threading.
5. Report Writing and Video Presentation

What learnt is most useful for you:

Through this project, I put the knowledge I learnt from IRS courses into practice, gained more experience in the overall process of Machine Learning model developing, including data(feature) collection, model assessment, data pre-processing, model training, deployment and usage.

Got more experience in combining model with system front-end, back-end and intelligent system design and optimizing.

How you can apply the knowledge and skills in other situations or your workplaces:

With more knowledge about Intelligent System developing, I am able to design an appropriate system workflow and select the most appropriate model to achieve the final purpose. Since I am responsible for model training and feature collection in the group project, in my future work, I can put all knowledge I learnt like cross validation, grid search for choosing best hyperparameters into practice.

What's more, in this project, we utilized the camera from laptop to capture and process images instantly, this skills would be useful in future work since there are many scenario in real-life requires image and video processing. Also, hand gesture recognition and classification can be used widely, it can help people to operate computer or other devices when people are not convenient to control keyboard and device, e.g. in bacteriological laboratory, operating room.

Name: **Xia Xu**	**Matriculation Number:** **A0261818Y**

Personal contribution to group project:

-Implement intellectual Q&A game and implement hand position analysis code
-Responsible for background removal algorithm and its result with OpenCV interface
-Use MySQL to create a Q&A questions database and make a visual-based interactive game
-Also help to write the reports, record the video, installation and user guide.

What learnt is most useful for you:

From this project, I learned how to use the MySQL database, and I find it really useful for data storage and searching. Also, I learned how to use python to modify the database and execute sql statements in python.
Besides, I also train the U-net with the dataset collected and have a good result. Thought at last we didn't use that model, I still learned how to train a neural network with my own dataset.
Now I know how to call the computer camera by OpenCV and use mediapipe to track the hands. With the result of processing the hands, I can develop more interesting functions.

How you can apply the knowledge and skills in other situations or your workplaces:

Whether on other projects or workplaces, I think there will be much more situations that need machine learning and database as these are the developing trend in this area. When I need to store data or create some tables, I know how to use the MySQL database. When I need to train a model to predict or classify something, I also know how to make the dataset and feed them to the net. Most importantly, in this project, I used many python packages that I have never used before. During I used them, I had to try to realize what the code was trying to do. I think this can improve my ability and comprehension of codes which will be very useful in any area.

Name: Kuang Shan	Matriculation Number: A0261839U

Personal contribution to group project:

System Design

I work on the system architecture to figure out a suitable structure for the project including choosing the language we are going to use to develop the frontend and backend modules and the data management.

Game module – Rocket Survival

he mediapipe from google can recognize people's hands give provide the screen coordinate for each finger. I use the hand joint coordinates extracted by mediapipe and use the relative position of the index finger to control the mouse cursor location to implement the game functions. Moreover, I worked with teammates to make experiments on different Machine Learning models to assess their performance on our project's gesture classification model and analyze their strengths & weakness. We together combine the model usage and deployment with the body part of our system code.

Frontend and Backend Development

In this project, I finished all the frontend and part of the backend work with my team. I am responsible for designing and implementing the frontend page structure, route, and functions. For the interface design, I developed every item users can see and interact with on the page. As for the functions, I develop all frontend functions that will handle users' actions such as click events and forward the data. For the route, I develop all the functions and modules that process users redirecting actions and forward the data. And for the backend, I work with my team to develop the functions that process the users' requests and related the FAQs database query.

Project Report

I wrote the user guide and all the related shell scripts and python scripts to help users to set up the python environment and initiate the database in an easier way. And also I work with my teammates to finish the project report including the part of the system I designed and the module I developed.

What learnt is most useful for you:

For this project, I learnt a lot about different parts of fields and they are all meaningful and useful for me. For system designing, I learnt how to choose the most suitable tools and languages. Sometimes the most popular languages or tools may not be the perfect choice to perform on some systems. All the factors should be taken into consideration. And for the project development, I have a better understanding of how to use Python to implement some functions and solve real-world problems. And for the model, first, I got some experience in how to carry out experiments to evaluate the performance of different models and how to analyze the performance of the model. And I learned to choose and use the most suitable models to implement the functions and solve real-world problems. And, I learnt how to deploy models to solve real problems.

How you can apply the knowledge and skills in other situations or your workplaces:

I am interested in the area related to system development and artificial intelligence, especially computer versions. After developing this project, I improved my programming skill which can quite boost my ability to solve coding problems, especially for Python. And I can apply my knowledge and experience gained from this project about model evaluation and deployment in the workplace to solve the model optimization problems.

Name: TONG SHIQIN	Matriculation Number: A0261970B

Personal contribution to group project:

1. Write the report, proofread the report.
2. Make a 5-min video (Complete materials (ppt, video, manuscript, etc.) for value, price, usedemo, etc.)
3. Complete the development of the fourth game, ready to train gestures to control the fourth game and embed it into our system, but canceled embedding it into our system due to the failure of gesture training.
   Introduction to the fourth game:
   Game character is a college student who has just graduated, and he feels unable to complete his tasks every day. Therefore, he hopes someone can help him complete the job in a limited time. We designed the game level time to be 200s, divided into multiple and continuous small time blocks, such as 20s as a time block. After every 20s, the player will return to the starting point and start again to collect the items to pass the level. At the same time, the player's behavior in all the previous small time blocks will act as shadows and help players to collect items together. With the help of the shadows, players need consider how to collect all the items before the last small time block. The specific video is in the attachment. We hope to show players through this game: Although there will be discouraged thoughts now (hope others help them), in the end, they will find that with their constant efforts (with the help of shadows), players can complete incredible tasks (collect all the items within the 20s or before the last small time block.
   CODE:
   https://drive.google.com/file/d/1vl6wEdHB6z2GEQOwcALL8MinR0OZ-Q2y/view?usp=sharing

   DemoVideo:
   https://drive.google.com/file/d/1TeCH7VtZt04cQ0WAk9NfBOS2G26cX7Bo/view?usp=share_link

What learnt is most useful for you:

I learned the complete project development experience, from GUI to algorithm. At the same time, I learned simple game (product) development: planning - business - development and implementation - artificial intelligence to embed(train gesture to control game); including but not limited to: how to train gesture processes; how to adapt to different environments (window and mac are different in running games); at the same time, it is necessary to estimate the achievable degree of the algorithm:(process of training gestures with unity game).

How you can apply the knowledge and skills in other situations or your workplaces:

1. Use gesture controls to complete more games.
2. Try our system's long-term vision plan: try to provide an interface for game companies to advertise their games. The specific description is as follows: "Imagine the following scenario: our software embeds more new games; it can be directly installed on various machines; then placed in shopping malls or other high-traffic places; users simply pass in front of it and have a direct opportunity to stop and play new games with simple finger interaction (avoid learning any game operation, allowing users to face the game itself directly); we believe that such a new type of game advertising has excellent value."

5.4 Appendix D: Games design

5.4.1 FAQ Game

Based on the region-detecting function, we come up with designing a FAQs game that uses finger gestures to get the answer. To increase the playability of the game, we decide to use live video streams to show the real-time movements of the users. To achieve this, we use OpenCV to capture the computer's camera. The cv2.videocapture class can process both video files and camera information in a very simple and fast way. And it can return every frame captured by the camera for image processing. The function is called as below:

```
import cv2
capture = cv2.VideoCapture(0)
while capture.isOpened():
    success, image = capture.read()
    cv2.imshow('image',image)
    if cv2.waitKey(5) & 0xFF == 27:
        break
capture.release()
```

Figure 13: Videocapture

In the while loop, we will instance the handprocess class and process each frame to get the result of position. When it comes to the game itself, there are two main tasks: the storage of the questions and how to show the questions. To store and access the questions, we decide to use the MySQL database for local use. MySQL has a wide range of applications because of its convenience and open-sourced codes. The structure of the dataset is like a tree. We create a database called questions and a table called QUESTION_tbl inside the database to store questions. The table contains seven columns including question_id, question, A, B, C, D and ANS. We can use the INSERT statement to add the questions one by one. But later we find this method is too slow and a waste of time. We then use the Navicat for MySQL, a browser for MySQL. After connect to the local host and then you can edit the tables very easily.

The storage of the questions has been finished, then we need to access the questions in the table using python. To achieve this, we need the pymysql package, which is a tool to connect to the MySQL server in python3. In this project, we just use the root user. Connecting steps are showing in the user guide. Beyond connection, we create a cursor with which the sql statements can be executed. Here the questions in the type of tuple contain all the questions, choices and correct answers. Then we create two empty dictionaries. One of

them uses the questions as the keys and the correct answer as the values. The other uses the questions as keys as well, but the values are the four choices. Therefore, later when we need to display the questions and check the answer, we can get the correct answers and corresponding choices by keys.

The whole program is continuous and in a while loop, so how to display the texts on the screen becomes a problem. As the function of adding text to the image is to modify the image matrices, so we can only use time as the trigger. In other words, we record the time when the program begins and display different texts on the image according to the time spent. For example, in the first 10 seconds, we display the first question and so on. During each conditional statements, we make a judgement if the user has given his choice. We define a buffer in the type of queue, and whenever there comes an action, the action will be appended and the queue's head will be dropped. As a result, if the set of the buffer is 1, then we can judge that the user has given his choice. Then we check the choice with the correct answer and store the score of the question to the list stated before. When the game begins, users will see the questions on the top of the window, and the four choices in four different areas and colour. Then the user just point at the choice he chooses, and the game will give information whether the answer is correct or not.

Figure 14: The FAQs game