Team 1: Navigation Helper Helmet

Capstone Project Report
April 16th, 2021

Client:
Abolfazl Razi: abolfazl.razi@nau.edu
GTA:
Han Peng: hp263@nau.edu

Team member:
Jingwei Yang: jy375@nau.edu
Alfred Ranasinghege Madushan Gunasekara: ag3232@nau.edu
Junlin Hai: jh2884@nau.edu
Bo Sun: bs968@nau.edu

Class: EE486C
Team Number: 1
Project Name: Navigation Helper Helmet
Date: April 16th, 2021

Contents

Class: EE486C
Team Number: 1
Project Name: Navigation Helper Helmet
Date: April 16th, 2021

Introduction:

a. Background of our Client:

Our Client is Dr. Razi. Professor Abolfazl Razi is an Assistant Professor of Electrical Engineering and Computer Science at Northern Arizona University. Dr. Razi's research laboratory is Wi-Fi and Smart Health (WiNeSh) research laboratory. His main research direction is AI-enabled networking, image processing, biomedical device design, ground, aerial robotics, and predictive modeling.

b. The problem being solved:

Vision is the main medium for people to obtain information about the external environment. People use glasses to observe the external environment to confirm information or avoid danger. However, many people in the world have a visual impairment. Due to visual impairment, they cannot live like normal people, and they even need others' help to complete some daily activities. These people are disadvantaged groups in everyday life, and there are ubiquitous dangers and difficulties in their lives. Therefore, to help these visually impaired, our Client and the four-team members wanted to design a wearable device to replace the user with basic information about the surrounding environment.

Our design is based on computer vision, which means that we want to use deep learning techniques such as object recognition and facial recognition to obtain basic information about the external environment. After the system obtains the basic information of the surrounding environment, we hope to describe the information to the user through the use of voice. We hope that users can understand the surrounding environment through voice. Therefore, our entire equipment is roughly divided into five subsystems: user interface, recognition system, LIDAR system, motor system, and voice prompt system. The main function of the user interface is to allow the user and the user's guardian to turn on/off the device and its functions or to set some parameters in the device. The recognition system is divided into two parts, face recognition and object recognition. Face recognition is to obtain the target person's name (if the target information is not included in advance, it will be marked as "unknown"). Object recognition is to determine the type of target object. The motor system will drive the LIDAR sensor and the camera to rotate together. Moreover, the orientation of the target relative to the user is determined by the rotation angle of the motor. The LIDAR system can obtain the distance of the target relative to the user. The voice prompt system can summarize all subsystems' information and then organize all the information into a complete and smooth voice according to a certain logical sequence. Finally, the information will be played to the user in audio form.

We hope that through the equipment we designed, users can understand the basic information about the surrounding change, including the location of obstacles, the types of obstacles, and the distance between them so that these visually impaired people can engage in simple daily activities and avoid danger to a large extent.

Class: EE486C
Team Number: 1
Project Name: Navigation Helper Helmet
Date: April 16th, 2021

Design Process:

a. Overall design process description:

Since our project's most important part is the object recognition system and the face recognition system, our group's first research is related to knowledge of recognition systems and deep learning. Since our group members do not have the corresponding professional knowledge, our main task in the fall semester of 2020 is to learn about object recognition and face recognition. Based on our simple trial plan and related knowledge study for a semester, we finally decided to use Python as the computer language environment. Python is a computer language that our team members are more familiar with. And our team has always believed that Python is easier to learn further. Then we also decided to use TensorFlow as our open-source platform for machine learning. The content of our research and the knowledge of learning we choose to use the object detection model is YOLOv4, and the algorithm for face recognition is HAAR.

Simultaneously, in the fall semester of 2020, we also screened and tested the hardware equipment we need to use. Through the communication between our team and customers, we learned that we need to use a more powerful microcontroller. Then based on our investigation, we decided to use Raspberry Pi 4B as our microprocessor. According to the basic requirements of the products, we need to design. Our team finally decided to use the other hardware, a camera, a LIDAR sensor, a motor, a power bank, a TF card, and a helmet.

Regarding the camera, we are using the Raspberry Pi Camera V2. The LIDAR sensor we chose is TF Mini-s. The motors we choose are 28BYJ-48 Stepper Motor and ULN2003 Driver. The memory size we choose to use for the TF card is 128GB. According to our test, these devices can usually be used. Then we still used a laptop with a GTX1070 graphics card for neural network training and sample processing. Next, we also completed the Raspberry Pi 4B and the computer to set the programming environment and operating environment. Our team has achieved recognition of tables, chairs, water bottles, monitors, and human recognition in object recognition after winter vacation. This winter vacation was a success because it significantly shortened the time required for our team to complete all tasks.

Then we started the formal design work in the spring of 2021. From January 11th to January 27th, we designed the voice system. What we use is to convert text to speech. Compared with human mechanically synthesized speech, there is no accent, and the speed is symmetrical, so it is easier to be accepted by most people. According to the voice system requirements, we have included many keywords, such as names of people, object types, length units, and location names. To make the voice that needs to be played smoother and smoother, we have also added many phrases, such as "There are a...", "and" "but" and so on. Then combine all the obtained keywords according to the language logic we have implemented and written.

The LIDAR system and motor system are also designed at the same time as the voice system. To make the information of the LIDAR sensor easier to be understood by users, we process the information obtained by the LIDAR sensor into the form of meters plus centimeters for storage. The motor system is extremely important in describing the orientation of an object. Because our team believes that it is unreasonable to use the degree of rotation angle as information to inform users because, as visually impaired, they are insensitive and unclear about the angle. Therefore, in order to make it easier for visually impaired users to understand the direction opposite to the obstacle, we decided to use the noun of the direction to describe it directly. These azimuth nouns include "front", "left", "right", "front left", and "front right". Therefore, we set the motor to rotate 45 degrees every time it rotates, and every 45 degrees corresponds to an orientation noun. For example, 0 degrees corresponds to "front," 45 degrees corresponds to "front right," and 90 degrees corresponds to "right." We think this way of description will be clearer for the visually impaired.

From January 11th to February 7th, we are also committed to completing and improving our object recognition system and face recognition system. For the object recognition system, we have also added the recognition of cars, bicycles, cats, and dogs. We performed more sampling and neural network training on group members' faces for the face recognition system and completed the target recognition. However, because the camera performance is not powerful enough, face recognition accuracy is always not high enough, especially for Asians, face recognition ability is poor.

From February 8th to February 21st, we combined the voice system, motor system, LIDAR system, and recognition system. Through the combination of these systems, we have further improved the voice system. After our thinking and practice, we let the voice system organize language by collecting information from other systems. If the target object is an obstacle, the voice system will first remind the user that there is an obstacle and then tell the user the type of obstacle, the obstacle relative to the user, and the distance between the obstacle and the user. For example, the voice system will say, "There is an obstacle on your right. That is a chair 1 meter 27 cm away from you." When a target is a person, the voice-prompt system will also tell the user the name of that person. For example, "There is an obstacle in front of you. That is a person called Junlin Hai, 45 cm away from you." But when the person is not someone you know, or you have not recorded his information in the system, the voice system Will say, "There is an obstacle in front of you. That is a person but does not know this person. He is 45 cm away from you." Our team believes that this description can clearly describe the surrounding situation to the user.

From February 22nd to March 26th, our team has continuously developed and improved our equipment's user system. Our user system is also developed based on Python, so it has good compatibility with our other scripts. Our user interface allows the user or his guardian to easily control our equipment, including modifying some of its parameters. We modify the parameters to let the user modify the parameters in the corresponding text file through the user interface and then let the script read the parameters in the txt file and replace them to achieve parameter modification.

In addition, because the customer repeatedly asked us to make a model to fix our hardware and helmet in the remaining time, we planned to use 3D printing to build the model initially. Still, because the model is too large, it takes a lot of time and money. We finally decided to use Acrylic Sheet as the material to make the model manually.

Class: EE486C
Team Number: 1
Project Name: Navigation Helper Helmet
Date: April 16th, 2021

b. Functional decomposition:

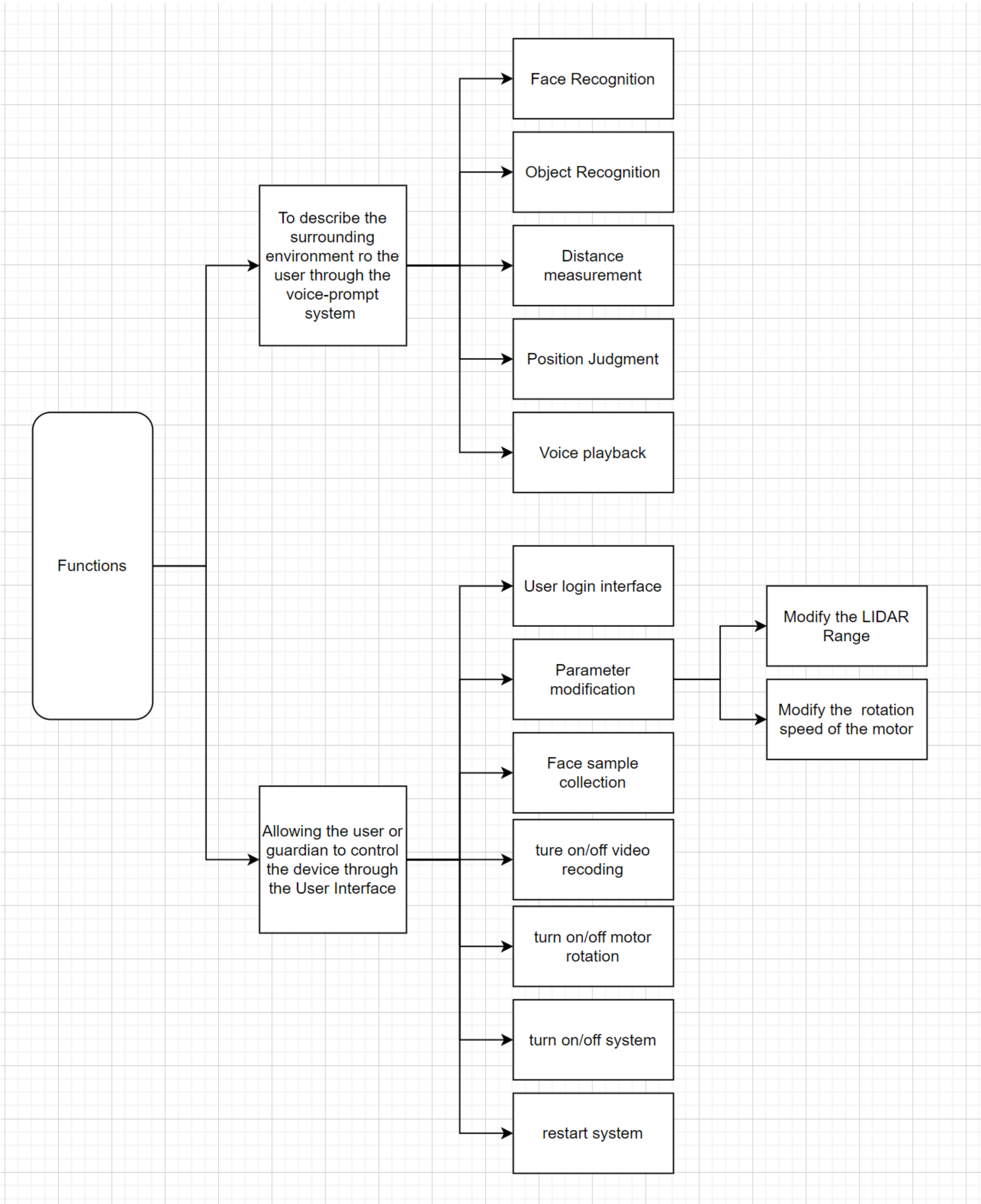


Image 1: Functional decomposition

According to the requirements given to us by our Client, our main functions can be divided into two. They describe the surrounding environment to the user through the voice-prompt system and allow the user or guardian to control the device through the user interface. The two main functions can be further subdivided into several small sub-functions. The first main function is to describe the surrounding environment to the user through a voice-prompt system. Its sub-functions are face recognition, object recognition, distance measurement, position judgment, and voice playback. The object recognition function and the face recognition function need to use the recognition system to complete. The distance measurement needs to use the LIDAR system. The position determination requires using a motor system, and the voice playback function requires the use of a voice-prompt system. The second main function has more sub-functions. The second main function is to allow the user or guardian to control the system and modify some system parameters through the user interface. Its sub-functions include user login interface, parameter modification, face sample collection, on/off video recording, on/off motor rotation, on/off the system, and system restart. When users use the login interface, they need to enter a pre-set username and password. In terms of modifying parameters, the user is allowed to modify the LIDAR monitoring range and the motor's rotation speed.

c. Prototype findings: results or effect

Our research and investigation of different subsystems and different functions in our project divided the prototype we need to design into two main parts. The first part is the hardware part, and the second part is the software part. The hardware part includes the LIDAR sensor, camera, and stepper motor. The software part includes a recognition system, voice conversion, and remote data transmission. Since the recognition system needs a camera, we combined the camera and the recognition system into a prototype.

The prototype is LIDAR, which is composed of Raspberry and LIDAR sensors. The main purpose of establishing such a prototype is to test the LIDAR system's feasibility, functional integrity, and LIDAR ranging function. For this reason, we used Python in the Raspberry Pi to write code that can drive the LIDAR sensor and tested whether it can measure distance. Our tests on the prototype found that the data obtained from the code conversion are all centimeters, so we converted these master data into meters plus centimeters. After testing, it was found that LIDAR can accurately measure the distance to the target when it runs alone. This also proves that it is possible to use LIDAR sensors for ranging.

The second prototype is the prototype of the motor. Our idea is to use this prototype to verify the feasibility of the motor driving the LIDAR and camera, the feasibility of controlling the angle of each rotation of the motor, and the feasibility of controlling the motor's speed. According to the survey, we found that the stepper motor needs to move 2048 steps per revolution, so every 45 degrees is 2048/8, which is 256 steps. So, we let the motor rotate 256 steps and then calculate it. We found that the rotated angle is 45 degrees, which also proves that we can control Martha to rotate 45 degrees every time and then pause for a while. We succeeded in preventing the motor's speed by changing the delay between every two steps of the stepper in the code. Therefore, we determined that we could drive the LIDAR sensor and camera by using the stepper motor as our driver.

The third prototype is about the recognition system. In this prototype, raspberry pi, a camera, and a laptop are used. First, we find on the Internet the code for object recognition that others have completed. Then we searched for 5,000 photos of cars on the Internet. After neural network training, we use a raspberry party with a camera to take pictures of the surrounding vehicles. Then we use the code to analyze the photos obtained by the Raspberry Pi. When the program thinks there is a car in the picture, it will output "Car." Therefore, when we see that the output has Car, we believe that the object recognition is correct. Since the code we choose to use is complete, it will output a decimal less than zero after outputting Car. This number represents the degree of certainty. For example, 0.98 means that the degree of certainty that the photo is a car is 98%. In the beginning, our confidence was low, so we collected more photos of cars. Then we conducted neural network training on these photos to ensure that the confidence level is greater than 85%. After testing this prototype, we are sure that we can realize the function of object recognition and face recognition.

The fourth prototype is speech conversion. This model is to test whether we can use computer-synthesized speech for speech playback. And the test system can automatically turn off the voice after the voice is played. In this prototype design, we used Google's GTTS for text-to-speech conversion. At first, we hoped to be able to implement the voice-to-text conversion. That is to say. We wish to use GTTS to convert information from different systems in real-time directly, then play them all in order, and finally delete these generated MP3 files. This is also to save the memory of the Raspberry Pi. However, due to the relatively slow calculation speed of the Raspberry Pi itself, there will be a delay in the real-time text-to-speech conversion in the Raspberry Pi. This also leads to frequent voices. It cannot play normally. Therefore, we decided to use the keywords converted by GTTS in advance to replace the real-time conversion process in the Raspberry Pi. With this modification, the Raspberry Pi can perform correct voice playback without delay and automatically turn off the voice.

The fifth prototype is remote data transmission. The significance of designing this prototype is to find a feasible way to exchange data between the Raspberry Pi and the laptop remotely. Due to the limited bandwidth and slow speed of the blue line transmission, we believe that using a wireless network to transmit data is undoubtedly the best choice. So the first thing we thought of was to use the laptop's wireless network to connect to the Raspberry Pi and then use FileZilla and samba (two kinds of connecting software using SSH protocol) to translate the documents. However, we also encountered some problems when using this method. First of all, if you do not continue to transmit data when you start these two programs, the SSH connection will be cut off after a few seconds so that data can no longer be transmitted. In addition, because these two are software, they need to be turned on every time you want to transfer data, which also causes a lot of trouble. Therefore, we chose other methods for data transmission, according to our tests and investigations. We plan to use a router to transfer data to transit stations. We let the Raspberry Pi and the notebook connect to this router simultaneously and then realize real-time data exchange through python scripts. To complete this prototype, we used the mobile phone as a router and realized the transmission between the Raspberry Pi and the laptop by connecting the mobile phone's hotspot. According to experiments, this method is feasible.

Class: EE486C
Team Number: 1
Project Name: Navigation Helper Helmet
Date: April 16th, 2021

Since our data transmission script selects the file to be transferred by judging the difference between the file in the Raspberry Pi and the file in the laptop, the file can be transferred in real-time, and the file with the same name can be replaced.

Class: EE486C
Team Number: 1
Project Name: Navigation Helper Helmet
Date: April 16th, 2021

Final Design:

a. System architecture with supporting details:

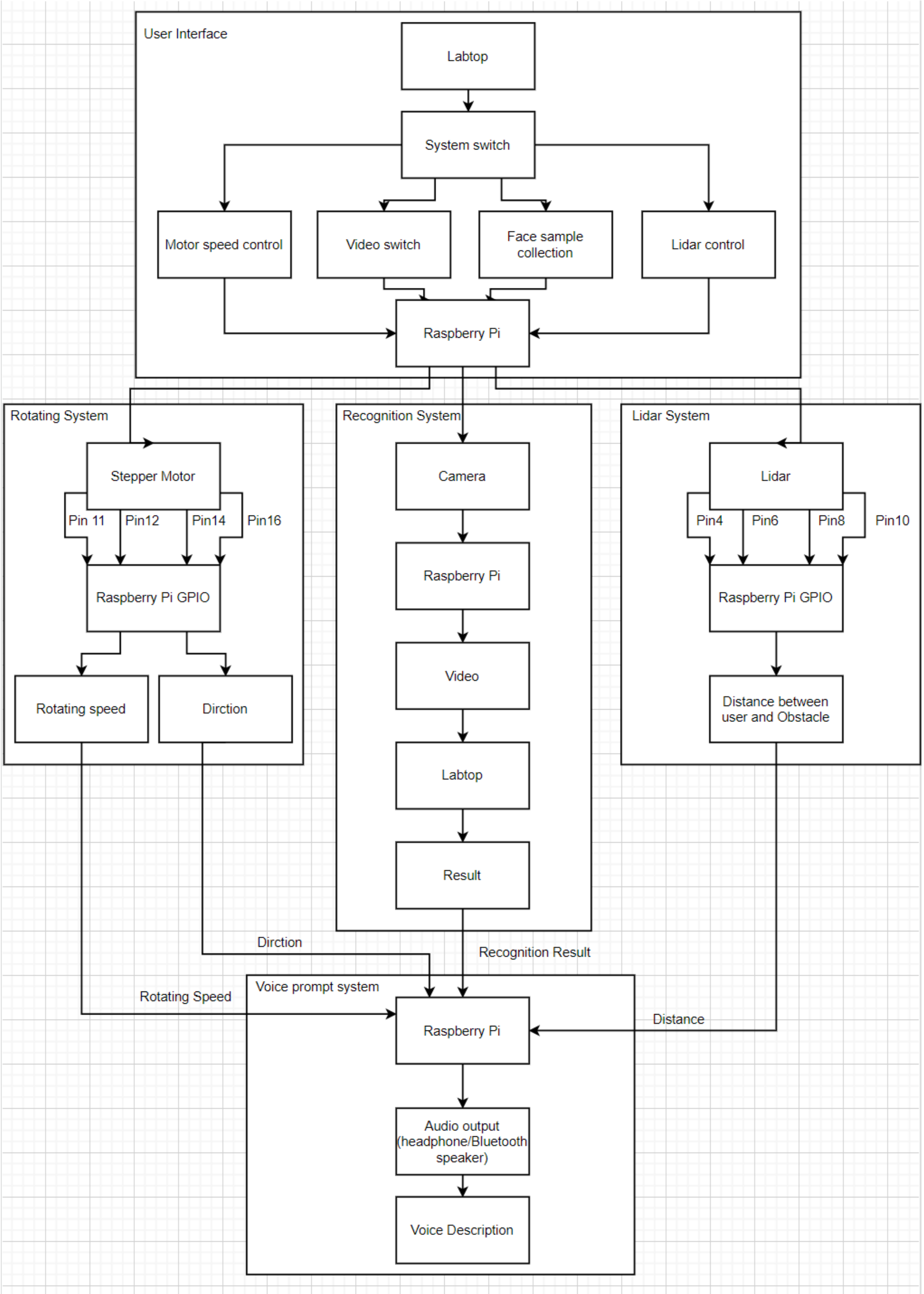


Image 2: System Architecture

According to our design and the results of research and testing, we divide the "system architecture" into five subsystems: the user interface, the motor rotation system, the recognition system, the LIDAR system, and the voice-prompt system. They cooperate to transmit data to complete the final task. In image 2, you can see how each subsystem works and how each subsystem ultimately outputs and transmits data.

b. Text explanations of the major components of the system:

According to our project's various requirements and constraints, the five subsystems shown in the system architecture in image 2 are all very important, and they represent one or more requirements or constraints. And the whole system can realize the final function only through cooperation between them.

It can be seen in the user interface of the subsystem that the data input port of the user interface is a laptop. Since our constraints indicate that we need to use laptops and Raspberry for remote data transmission, this design is in line with our products' constraints. Then, to meet the Client's requirements, for example, the user can use the user interface to control the detection range of the LIDAR and the speed of the motor. The user interface we designed allows users to change these parameters in a simple way. After the user inputs these parameters, the laptop will transmit these parameters to the corresponding text file in the Raspberry Pi. When other scripts are running, these files will be read, and the data in these files will be applied to control the system and modify parameters.

In the motor rotation system, you can see the connection between the Hibiscus machine motor and the Raspberry Pi and the output data from image 2. According to the Client's requirement, we need to use a motor to drive the camera and the LIDAR sensor to rotate. Therefore, such a design meets the constraints and requirements. According to the Client's requirement, our voice system needs to be able to describe the position of the object relative to the user. Therefore, the motor rotation system we designed can output the direction, which is based on the angle of the motor rotation. For example, when the motor rotates to 0 degrees, we think that the obstacle detected at this time is directly in front of the user. When the motor rotates to 45 degrees, we think that the detected object is in the user's front right. And this orientation will not be changed by the user's west-east or turn because our device is a helmet, the user always needs to wear the helmet on his head, so when the user moves, the LIDAR and camera will move with the user.

According to constraints, in the LIDAR system, we need to use at least one LIDAR sensor to measure the distance between the obstacle and the user. Therefore, the design of our system architecture meets the constraints and requirements. Compared with other systems, LIDAR is the simplest. It only needs to measure the distance between the user and the target object, then convert these data into meters plus centimeters, and store the data in a file called meters.txt In the file and centermeter.txt file. These two files will be used to judge the distance, and the distance file will be called by the voice system. In addition, these two files will also be compared with the LIDAR detection range set by the user in the user interface. When the value in the file is greater than the user's value, the voice system will not broadcast the currently detected object. When the file's value is less than the user's detection range, the voice system will describe the object by voice. This completes the requirement that the user can set the LIDAR monitoring range. Although the detection range of LIDAR has not changed in fact, the distance described by voice means that objects beyond the range will not be described, which is equivalent to modifying the detection range of LIDAR.

We need to use at least one camera to design the recognition system according to constraints regarding the recognition system. So in our system, we use a Raspberry Pi camera. According to demand, our recognition system should be able to perform object recognition and face recognition. Therefore, in our system architecture, we use a raspberry pie and a camera to take a video, then transmit the video to a laptop in real-time, and then use the laptop to recognize and process the video. This process can be seen in image 2. Our team aims to speed up the calculation speed of the recognition system and make it easier to monitor the program through the laptop. In the recognition system, object recognition and face recognition are not performed at the same time. Face recognition will only be performed when the object is recognized as a recognized object. This is also to reduce the bug of the voice-prompt system. Therefore, on the monitor screen of the laptop, you can see two boxes appearing at the same time (object type and the name of the person. A complete schematic is shown in appendix 1).

The last major component is the voice-prompt system. According to our Client's constraints, we need to use a voice system to describe the objects detected by the device. Therefore, we have designed such a voice system. We will use GTTS to convert keywords into voice files (.mp3 files) in advance. Then, match the item pair's voice files according to the information collected from other systems, and then organize these voice files into a paragraph according to the language logic that has been set. This saves the internal use of GTTS every time it needs to be transferred and greatly reduces bugs. As shown in image 2, the voice system collects the position information from the motor's rotation, the recognition result from the recognition system, and the distance information from the LIDAR system. Organize this information into a complete and smooth sentence. For example, if there is an obstacle in front of your left, 1 meter 32 cm away from you, that is a chair. But there is one inside and outside. When the user is in front of a wall, the voice system will only say that there are obstacles in front and the distance between the account and the user.

Class: EE486C
Team Number: 1
Project Name: Navigation Helper Helmet
Date: April 16th, 2021

Result:

a. The requirements spreadsheet with the color indication of requirements testing results:

Type of Test	Status	Req #	Requirement
		1	Recognition System
Integrate	Finish*	1.1	Object recognition for a variety of common objects. (table, chair, bottle, tree, Car, bicycle, train, monitor, cat, dag)
Integrate	Finish*	1.2	Face recognition of familiar people.
		2	Rotating System
Inspect	Finish*	2.1	Can drive the camera and LIDAR to rotate together.
UTM	Finish	2.2	The stepper can swing left and right. And the angle range is +90 degrees to -90degree
UTM	Finish	2.3	The motor rotates 45 degrees each time.
		3	LIDAR System
UTM	Finish	3.1	Can detect distances from 0 to 400 cm.
UTS	Finish	3.2	The error does not exceed 5cm
		4	Voice Prompt System
Integrate	Finish*	4.1	Object: It can describe the type of object, the distance, and direction relative to the user.
		4.1.1	When the object recognition system's object recognition system is an unrecorded object, the voice-prompt system will still prompt the user that there is an obstacle.
Integrate	Finish*	4.2	Person: It can describe whether the line being detected is a person familiar to the user.
		4.2.1	When the detected object is a person familiar to the user, and the sample has been collected, the voice broadcast explains the detected object's identity. If the detected object is not recorded, the user needs to be notified that it is an unknown person.
Integrate	Finish	4.3	Describe the direction by the angle of rotation of the motor.
Integrate	Finish	4.4	The distance between the object and the user can be described according to the measurement result of LIDAR.
		5	User Interface
UTS	Finish	5.1	The user can adjust the range of LIDAR detection. The adjustable range is 30 cm to 300 cm.
UTM	Finish	5.2	The user can adjust the speed of the motor.
Inspect	Finish	5.3	Allow users to turn on or off the video recording function.

Inspect	Finish	5.4	Allow users to turn on or off the motor rotation function.
Inspect	Finish	5.5	The user needs to use a username and password to use it.
Inspect	Finish	5.6	Allow users to turn on, turn off or restart the device through the user interface.

Table 1: The requirements spreadsheet

In Table 1, the black parts are requirements that do not need to be tested, the green parts are requirements that completely pass the test, and the yellow parts are requirements that basically pass the test. Then the content marked with an asterisk is the most important requirement.

b. Important test results:

Test 1: (A complete schematic is shown in appendix 2)
Test Case Name: The angle of each rotation of the stepper motor.
Tester: Alfred Ranasinghege Madushan Gunasekara
Date: February 22, 2021

In the one-time test, the tester tested the requirement 2.2 "The motor rotates 45 degrees every time". The purpose of this test is to determine whether there is an error in each rotation angle of the motor and whether the error is acceptable. Our team believes that this test is a white box test because we can predict the output result based on known data and equations.

The specific experimental steps are as follows. First of all, we made sure that the stepper motor model we used was 28BYJ-48, and then we read the manual of this stepper motor. We know that every revolution of this motor needs to go through 2048 steps, which means that the motor can move for one week after each electrode changes 2048 times. Based on this, we can calculate that if the motor rotates 45 degrees, the stepper motor needs to be changed 256 times, that is, 256 steps. To accurately calculate and understand the number of steps the motor has moved, we have added a counter to the code of the motor rotation and set it to increase by one after each change of the electrode. Each time the counter increases by 256, the electrode will stop for 30 seconds. The tester will measure the angle that the motor has rotated at this time. According to the difference between the actual result of each measurement by the tester and the ideal result obtained according to the calculation, we can get the error of the motor rotation angle. Considering the motor itself's error and the error that the surveyor will produce when measuring, our team believes that when the angle error is within +/-5 degrees, it is considered an acceptable error. The specific test results are shown in Figure 1.

According to this test, we know that every rotation of the motor will have an error, but the error is less than 5 degrees. Therefore, such an error is acceptable, so we believe that this time the test is passed.

Class: EE486C
Team Number: 1
Project Name: Navigation Helper Helmet
Date: April 16th, 2021

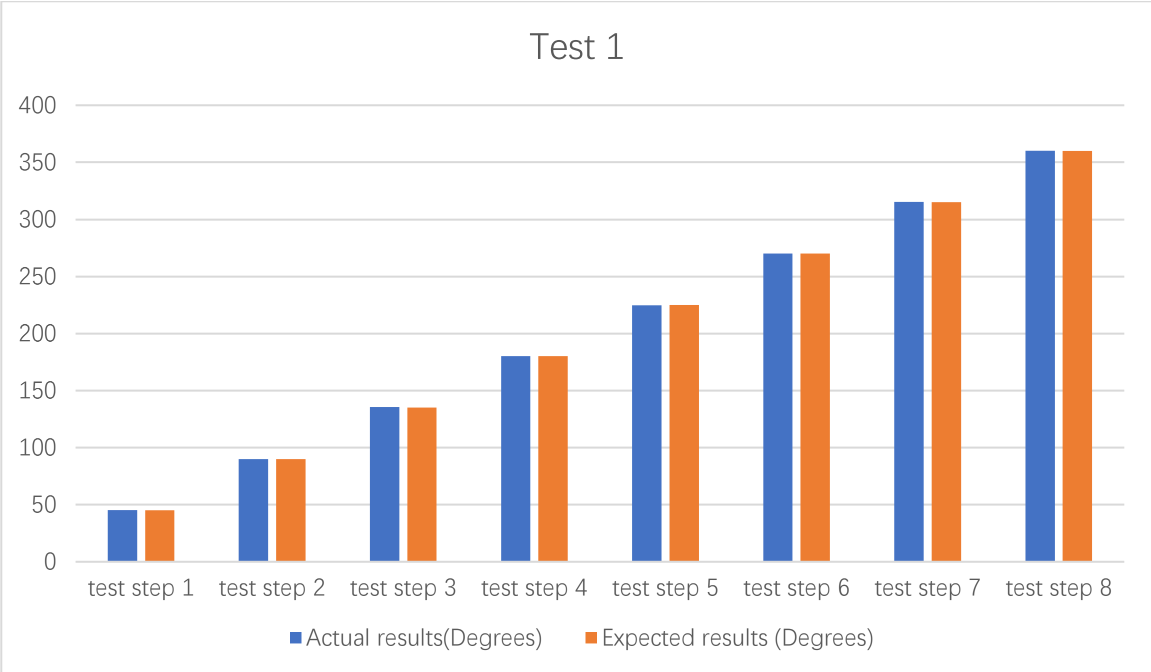


Figure 1: Result of test 1

Test 2: (A complete schematic is shown in appendix 3)
Test Case Name: Motor speed control.
Tester: Jingwei Yang
Date: February 24th, 2021

In order to test requirement 5.2, "Users can adjust the rotation speed of the motor," we designed and implemented this test. According to requirement 5.2, we should allow users to adjust Madian rotation speed through the user interface. According to the motor's user manual, we learned that 28BJY-48 needs to move 2048 steps per revolution. Therefore, according to this feature, we add a delay of 1 millisecond to each step. Therefore, it can be concluded that theoretically, the time required for each revolution of the stepper motor is 2.045 seconds. Then we divide the experiment into 12 steps. The first step is only when the delay is set to 1 millisecond, and each subsequent step will increase the delay by one millisecond. Then we will subtract the theoretical cycle time and the actual cycle time minus real-time to get the error value. Our team believes that the error is acceptable when the error value is less than 1 second due to the mechanical error and measurement error. This time the test input and output are both numbers, so we use unit test (matrix).

According to the test results, step 1 failed (see Figure 2) because the motor cannot normally rotate when the delay is a few milliseconds. The delay error of other steps did not exceed 1 second. But after our test, we found that when the delay is greater than five milliseconds, the motor will shake during the rotation. Therefore, we set the number of delays that the user can control as two milliseconds, three milliseconds, and four milliseconds. Correspondingly, I set them to 3 options "normal," "slow," and "very slow."

Class: EE486C
Team Number: 1
Project Name: Navigation Helper Helmet
Date: April 16th, 2021

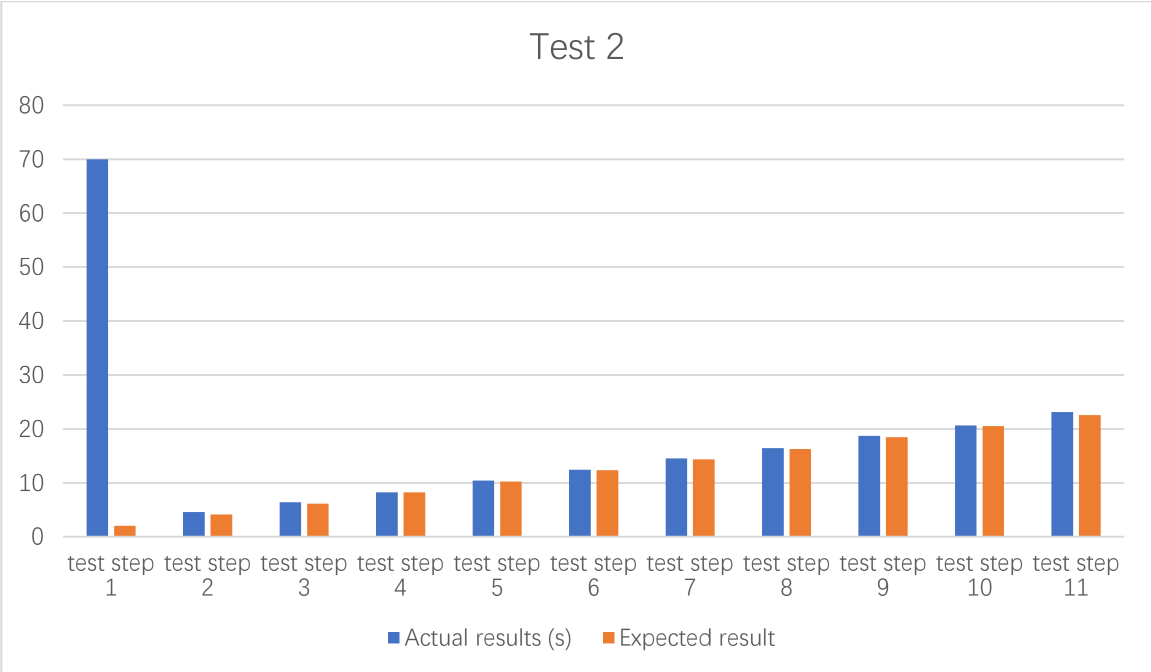


Figure 2: Result of test 2

Test 3: (A complete schematic is shown in appendix 4)
Test Case Name: LIDAR control function detection
Tester: Bo Sun
Date: February 24th, 2021

What this test needs to detect is that 5.1 users can adjust the LIDAR monitoring range through the user interface. It is very important that users can freely adjust the monitoring range of LIDAR. Because this can control the frequency of the voice-prompt system, it can reduce a lot of unnecessary interference.

In this experiment, we first place a chair in any position. Then we need to use a ruler to measure the distance between the chair and the user (LIDAR sensor). Next, we need to adjust the LIDAR monitoring range. If the LIDAR monitoring range we set is smaller than the actual distance between the chair and the user, then the program will output "Out of the Range." through the calculation of the script. If the range we set is greater than the distance between the chair and the user Distance, then the system will output the distance it measured. Then I will use this data to compare with actual data. This experiment requires constantly changing the distance between the chair and the user and the LIDAR monitoring range. So as to simulate the result of the user changing the LIDAR monitoring range under different conditions. We also tested the error of LIDAR in detecting the distance between the object and the user in this experiment. We also judge whether LIDAR passes the test by judging the size of the error. Our team believes that due to our manual measurement of the distance between the target and the user, there will be errors. And the LIDAR sensor will also produce systematic errors when measuring. Therefore, we believe that if the difference between the measured value and the predicted value does not exceed 5cm, then we consider the test passed.

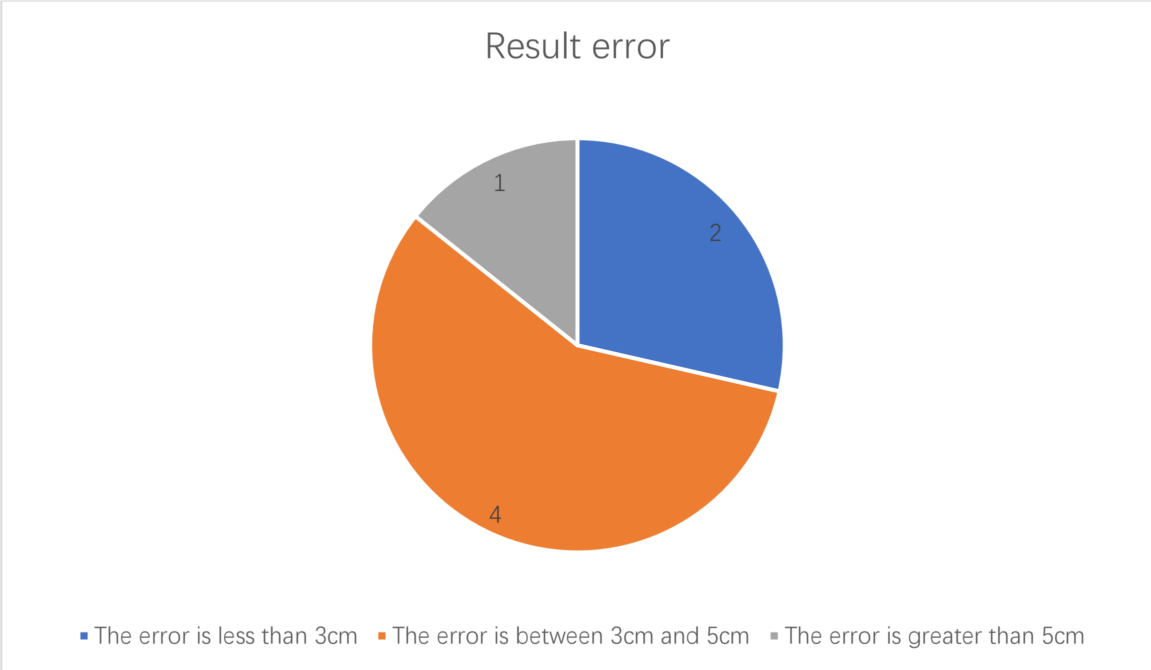


Figure 3: Result of test 3

Test 4: (A complete schematic is shown in appendix 5)
Test Case Name: User integration testing
Tester: Junlin Hai
Date: February 25th, 2021

This test's primary purpose is to determine whether the voice system can accurately describe different people's descriptions in different locations. Its corresponding requirements are: "1.4 Face recognition of familiar people. 4.3 The voice system can use voice to describe whether the user's task is familiar to them. 4.4Describe the direction by the angle of rotation of the motor. 4.5 The distance between the object and the user can be described according to the measurement result of LIDAR." Because we understand the internal organization of the system used in this test, and we can predict its results. Therefore, our group considers it a white box.

Through face recognition of different people, the voice prompt system will perform different voice descriptions. And according to the person's location and distance, the voice system will also change the words used in the book to ensure that the description is more accurate. This test will use more than three different systems to complete a specific function. Therefore, we consider it an Integration Test.

We used four different detection objects in this test: Jingwei, Bo, Alfred, and Jingwei's roommate. First, we let Jingwei stand in a different position to be detected. We anticipated and outdated, according to the location of Jingwei, different voice systems will also change the description of the direction and the description of the distance. Then we describe different people. Since Jingwei, Bo, and Alfred are all familiar people, the voice system will indicate their names. But Jingwei's roommate is not recorded, so the voice system will notify you, "You don't know this person." From the second step to the sixth step, the test's main content is to detect whether the voice system can accurately describe the detected object's location. The first step to the ninth step is to test whether the voice system can accurately distinguish between familiar people and strangers. Due to the LIDAR measurement errors, the actual results may deviate from the predicted results in terms of distance description. But if the error is within 5 cm, we consider it passed.

c. Analysis of Result:

In the test conditions, all of our tests have basically passed the requirements in addition to LIDAR detection for objects that are too small, such as water bottles. There was an error that exceeded expectations. Because hardware conditions limit part of this error, our team considers this to be an acceptable error. Also, most of the test results are what we expected. Our five most important requirements have all passed the test. Although there are still shortcomings, they have basically met the requirements for accuracy and stability.

Class: EE486C
Team Number: 1
Project Name: Navigation Helper Helmet
Date: April 16th, 2021

Conclusion of Capstone Report:

a. Most important requirements and their results:

1.1: Object recognition for a variety of familiar objects. (table, chair, bottle, tree, Car, bicycle, train, monitor, cat, dog):

This requirement is the most important and basic requirement of our project. Because its realization proves that our products can be used, we conducted corresponding tests. For our test, the result is that we can identify many different items. Moreover, when the object is in an environment with relatively good lighting conditions, high-precision recognition can be achieved. For tables, chairs, drinking glasses, and displays, the recognition rate of common objects is usually 90%. But when the light conditions are relatively poor. The recognition rate will decrease.

1.2: Face recognition of familiar people:

This one is also the basis of all the functions of our project. Therefore, to realize the recognition of human faces, we use a large number of facial photos of internal members of our group as samples. However, after various tests, we found that the facial recognition system is greatly affected by light and angle when photographing faces. When the light is sufficient, the recognition accuracy is high, and when the light is dark, the recognition accuracy is poor. According to many of our tests, we found that the facial recognition system has a poor ability to recognize Asian faces. For example, recognition systems often confuse latitude and longitude with Bo.

2.1: Can drive the camera and LIDAR to rotate together.

This item requires the motor to be able to drive the LIDAR and the camera at the same time. It is very important for customers. Our equipment mainly determines the direction of the target relative to the user through the motor's angle of rotation. Therefore, the motor's synchronization, LIDAR, and the camera can ensure an accurate and detailed description of our equipment's surrounding environment. When this item is not satisfied, the user will not know the first position between the detected object and the user. This will cause the user to be unable to know the target's exact location even after learning the object's type and information, so the user cannot avoid obstacles or dangers. This is very dangerous, so this requirement is very important. According to this requirement, there is no problem after the test. The stepper motor can drive the LIDAR sensor and the camera to move together.

4.1: For the object, it can describe the type of object, the distance, and the direction relative to the user.

This requirement is the basic way to provide customers with a description of the environment. Since our users are visually impaired, it is the only way to describe the user's environment through voice. Therefore, it is very important for users. When this requirement is not met, the user will not know any information about the surrounding environment. Even if our equipment has identified and positioned the surrounding environment, users still cannot obtain any information. This item's result fully meets the requirements, and the voice-prompt system can make accurate voice descriptions based on the information obtained.

4.2: For the person: It can describe whether there are familiar people around the user and also describe familiar people's names.

This requirement is equally important because audio is the only way for our users to understand the surrounding environment. If this requirement is not met, the user will not know whether the identified person is familiar to him. This will affect the integrity of the functional basis of our equipment. Our final design fully meets this requirement. The voice-prompt system will announce the name of the task based on the result of face recognition. And when the target person is a stranger, the voice system will tell the user, "you don't know this person."

b. Lessons Learned:

Through testing, we have learned more effective ways to find and solve problems. When we do not know how to improve the product or do not know the product's current defects, we can use the method of system testing to find the products that need to be improved and the errors that need to be avoided. After we modified the UI function, we conducted regression testing. The test also successfully helped us avoid trouble. Since we forgot to set the initial value of the motor rotation angle after the modification, the motor rotation cannot be reset, which also caused other systems to fail to start normally. Through regression testing, we checked the modified location and finally found the problem. We found that the file data that recorded the number of motor steps were lost. When we re-enter and give it an initial value, it can re-rotate. This has taught us how to find problems.

According to all the tests and repairs to these results, we found some problems that we could not solve well. We found that the accuracy of face recognition in our recognition system is too low. Moreover, when LIDAR is used in conjunction with a camera, LIDAR's measurement accuracy cannot meet the requirements. Simultaneously, due to the hardware system's limitations, the system response time and startup time are too long. Therefore, in order to solve these problems, we propose some hypothetical solutions. As for face recognition accuracy, we think we need to use a more powerful USB camera or wireless camera instead of a Raspberry Pi camera. This can improve the sharpness of the camera shooting and pictures, thereby improving the recognition accuracy. We can also use more samples for neural network training to improve the accuracy of recognition. In order to improve the accuracy of LIDAR measuring distance, we can use two LIDARs to measure the distance at the same time. When the distance error detected by the two lidars is within 5 cm, two non-parallel lidars will be used to detect the same object and calculate the distance between the user and the target. This can reduce measurement errors. Since two lidars are used to measure the same object at the same time, and the difference between the two-measurement data cannot be greater than 5 cm, it can be determined whether it is the measurement of the same object or the measurement of a plane, to improve the response speed and startup speed of the system. We can use more powerful microprocessors.

Class: EE486C
Team Number: 1
Project Name: Navigation Helper Helmet
Date: April 16th, 2021

User Manual:

a. Introduction

Dear Client, first, thank you very much for choosing our group "Eye of the Future." We are very honored and happy to be able to provide you with services and help solve problems. I want to thank you from the bottom of my heart and hope you will like our products. Just as this product is required, this product should have an identification function to replace the human eye to complete some information collection work, for example, to identify some daily necessities and familiar people's faces. This product should also have a monitoring range of at least 180 degrees to ensure the user's safety and the accuracy of the description of the user's surrounding environment. This product is also required to have a voice system. That is to let users better understand the information of the surrounding environment through voice. This will also be the main way we present the functions of our product to users. To meet these needs, our team has designed several functions for the product "Navigation Helper Helmet":

1. The product can perform face recognition and object recognition.
2. The motor that accompanies you in the product can drive the LIDAR and the motor to swing together.
3. The product has a powerful user interface.
4. The product can provide voice prompts to users.
5. The product can measure the distance between the user and the obstacle.

These functions will be explained in detail in this user manual. We only make this user manual to help you use this user manual more smoothly in the future and to repair this product successfully. We hope this product can give you more help in the future.

According to the project requirements, the product should have an identification function to help users come to the information in the mobile phone's surrounding environment. Therefore, we believe that the most important issue is that this product should collect environmental information. This means that we need to find and use the right ways to help visually impaired people obtain environmental information. Also, it is mentioned in the requirements that we need to use to drive LIDAR and the camera to detect at least 180 degrees. This shows that the product's choice of hardware and the choice of hardware will be significant. And because it is essentially different from the previous requirements, he should be assigned to work in other systems. The following requirement is that our products should have voice capabilities to describe the environment around the user. This means that we need to summarize all the information to organize them into a complete sentence and broadcast it to the user by voice. To ensure that the voice prompt is as clear as possible, the voice should be artificially synthesized.

According to the various requirements and regulations put forward by users, we divide our products into five sub-systems according to the different functions that need to be realized: recognition system, motor rotation system, LIDAR system, voice-prompt system, and user interface. Among them, the sub-system "recognition system" can help users collect environmental information and is established, it can complete the requirements for object and face recognition. Therefore, it is also divided into two parts: face recognition system and object recognition system. The face recognition system can recognize people familiar to the user and distinguish their names, and the object recognition system can recognize some common object types in daily life. The motor rotation system meets the requirement that the motor can drive the LIDAR and the camera to rotate. The motor rotation system should be more able to cooperate with the recognition system and the LIDAR system more stably. And record the new information about the motor's rotation and sending it to the voice system to ensure that the voice system can use this information. The LIDAR system consists of only one LIDAR sensor. Its main function is to use the LIDAR sensor to detect the front and determine the position between the obstacle and the user. The same as the motor rotation system is that the lidar system will save the information and let the voice system use it. To complete the requirements, we also need to establish a voice-prompt system, which is to achieve the requirement that the system can describe the surrounding environment to the user through voice. It is an information collection station and processing center. All the information from other systems should be gathered here, and the voice-prompt system will organize them into complete words to be played to the user. Since the target users are mostly visually impaired, to help the user's guardian or assistant operate the device more conveniently, we have also created a user interface. The client and we believe that the user interface should control and manage the entire system. He should be able to effectively control the motor system, recognition system, and LIDAR system. These three systems are systems for obtaining information about the external environment, so users should be able to customize them. We hope that the motor rotation system, LIDAR system, and recognition system can work in an orderly manner under the user interface's control. Then the voice-prompt system will further process all the information to use voice to describe the surrounding environment to the user. [A complete schematic is shown in appendix 6]

To achieve all the above systems and better fulfill all the requirements, we have launched a series of research and discussions. According to our meeting, we have evaluated the four most important issues. The first thing that needs to be is the construction of the recognition system. The judgment of the obstacle position, then the judgment of the distance between the obstacle and the user, and finally, it integrates all information and the description of users through the voice system. First of all, to complete the recognition system, we decided to use YOLOv4 as the object recognition system's monitoring model because this system is more mature and stable. We also decided to use HAAR as the basis for creating a face recognition system, which should be able to detect human facial features well. To better detect and process the image, we used a wireless transmission system to transmit the video obtained from the Raspberry Pi to the computer in real-time for detection. This is done to speed up the image processing. Then we refer to the motor angle to judge the obstacle's orientation and divide the orientation into five, namely: front, left, right, left front, and right front. It is divided evenly according to the range of motor rotation from 0 degrees to 180 degrees, which means that there is a difference of 45 degrees between each orientation. Doing so can avoid the effects of the user's head-turning. For the distance between obstacles and users, we use LIDAR sensors to form a LIDAR system. Then we convert the obtained data into meters plus centimeters (for example, 1 meter

29 cm). We do this to allow the voice system to describe the distance more clearly. Google's GTTS generates all the voices in the voice system we use. We use artificially synthesized speech because we believe that artificially synthesized speech has no accent, so it is clearer to most people. The voice system recovers the information from all other systems and organizes them into a complete and fluent sentence. This sentence will include the type/name of the object/person, the position relative to the user, and the distance between the user (For example: "There is an obstacle on your left, 1 meter 45 cm away from you. That is a chair." or "There is an obstacle on your right, 23 cm away from you. That is a person, he is Lee."). For the user interface, it is not necessary but also very important. Our team is based on Python to write him. It has the functions of adjusting the LIDAR monitoring range, adjusting the motor speed, turning the motor on or off, turning it on, turning off the device, recording video, adding facial recognition samples, and restarting the system. The user interface is a control platform designed for users, so every time you log in, you should enter your username and password to ensure safe use. Through this test, we have realized that our equipment can complete all functions in a stable environment.

b. Installation:

Hardware Part:

- Raspberry Pi 4B:
  The rated voltage of the Raspberry Pi is 5v. It needs to use type-c to charge first. You can use a wired network or use Mini-HDMI to connect to the computer.
- Camera:
  Please connect the camera to the Raspberry Pi through the CSI interface, and be careful not to use too much force to avoid damage to the lock.
- LIDAR Sensor:
  The LIDAR sensor has a total of 6 wire structures, which need to be connected to Pin4, Pin6, Pin8 and Pin10 of the Raspberry Pi GPIO.
- Stepper Motor:
  The stepper motor needs to be connected to its driver first, and then pin 1, 2, 3, and 4 of the drivers is connected to pins 11, 12, 14, and 16 of the Raspberry Pi GPIO. To ensure that the connection is correct, please observe whether the four red LEDs on the drive light up in sequence during the test.

Software Part:

Computer

- Install Anaconda:

1. First, log on to the website https://www.anaconda.com/distribution/ to download the new version of Anaconda.
2. Then select 64-Bit or 32-Bit to install according to the type of computer.
3. Next, choose Install for "Just Me" or "All Users" according to your needs. I recommend choosing "Just Me" for this step.
4. Then check all the options in Advanced Options. This is for ease of use.

5. Finally, click "Install" to complete the complete installation.
6. The installation time will vary depending on the computer configuration, please be patient.

- Install cuDNN and CUDA:

1. First, log on to the website: https://developer.nvidia.com/cuda-10.0-download-archivetarget_os=Windows&target_arch=x86_64&target_version=10&target_type=exelocal
2. Then, log on to the website https://developer.nvidia.com/cudnn to download cuDNN7.4.1.5
   You will get two folders as shown in image 3.

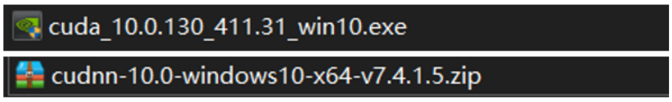


Image 3

3. Open the cuda.exe file to start the installation. Please select custom in the installation options. Then click Next.
4. In the custom installation option, please select according to image 4, and then click Next to install. After installation, you can find the root directory at this location on the C drive. (C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.0)

Image 4

5. Then unzip the Cudnn.zip to get those file shows in Image 5. Just copy the contents directly to the root directory of the C drive (shows in Image 6).

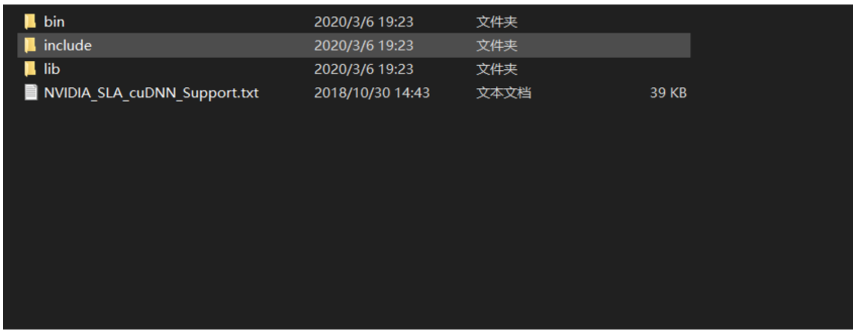


Image 5

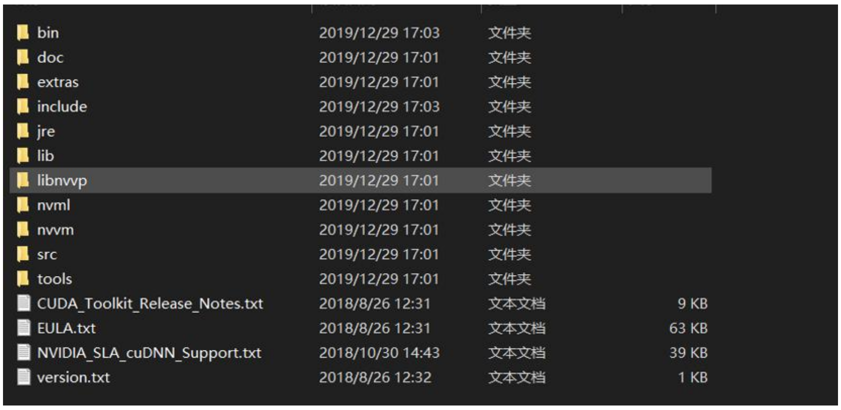


Image 6

6. This will complete all the installation.

- Configure TensorFlow environment:

1. Press Win+R start cmd.
2. Enter the following command in the command prompt:

```
conda create –n tensorflow-gpu python=3.6
activate tensorflow-gpu
pip install tensorflow-gpu==1.13.2
pip install keras==2.1.5
pip install h5py==2.10.0
```

3. After all installations are complete, restart the computer.

- Set TensorFlow as default environment:

1. Simply right click on the ‘This PC’ icon. Then, select ‘Properties’.
2. Next, click on the ‘Advanced system settings.’
3. Next, click on the ‘Environment Variables…’ (As shown in the image 7)

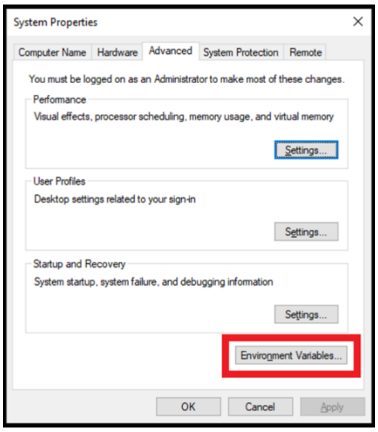


Image 7

4. Then, click on 'Edit…' to the variables named 'path'.
5. Delete all the other variable value in it and add new value in the TensorFlow folder. For example, as shown in image 8.

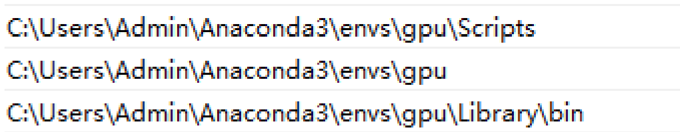


Image 8

6. Then, change the path from gpu to tensorflow-gpu.

Raspberry Pi

- Install the Raspbian system (Raspberry Pi Imager):

1. First log in to the website https://www.raspberrypi.org/downloads/ and download the version you need. For example, we are using Windows, so we need to download the Windows version.
2. Then, insert SD card on your computer.
3. Run the previously downloaded Raspberry Pi Imager and choose Operating System.
4. Choose Raspberry Pi OS.
5. Please make the next settings according to your personal needs.
6. When the installation is successful, click "choose storage". Then choose SD card.
7. Then click "write" and let system write and verify SD card.
8. Then install the SD card into the Raspberry Pi, so that the environment of the Raspberry Pi is installed.

- Set raspberry pi wireless connection:

1. First, create a blank text file named "wpa_supplicant.conf". Fill in the content according to the following reference format and save.

```
country=us
update_config=1
```

```
ctrl_interface=/var/run/wpa_supplicant

network={
 scan_ssid=1
 ssid="MyNetworkSSID"
 psk="Pa55w0rd1234"
}
```

2. Both the SSID and password should be surrounded by quotes.
3. Create a new blank file and named it 'ssh' (LOWER CASE AND WITHOUT ANY EXTENSION NAME)
4. Copy the two files to the boot partition on your SD card's boot folder.
5. Then, install putty: https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html
6. Next, download the 64-bit version. (As shown in image 9)

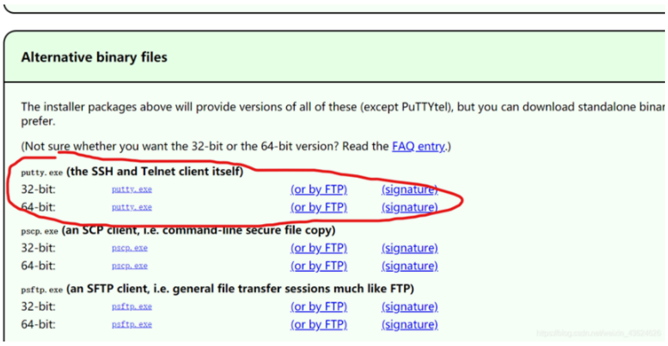


Image 9

7. Open putty, enter raspberrypi, then press open. (As shown in image 10)

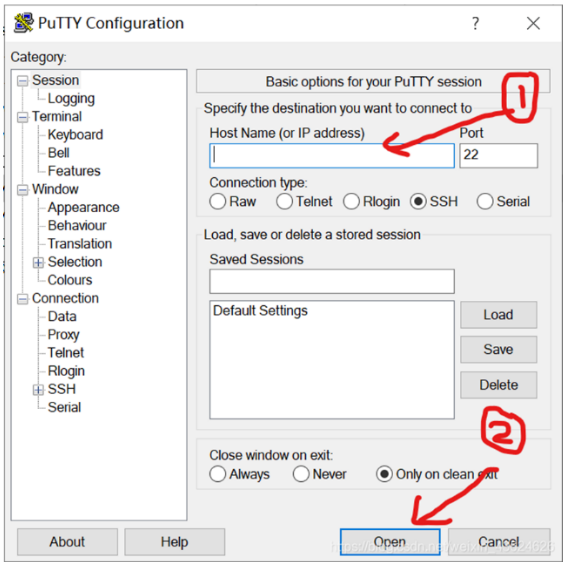


Image 10

8. Next, enter the yes.
9. Then enter the raspberry pi's account and password (account:pi password:raspberry)

[it would not show up anything when you type in password] (As shown in image 11)

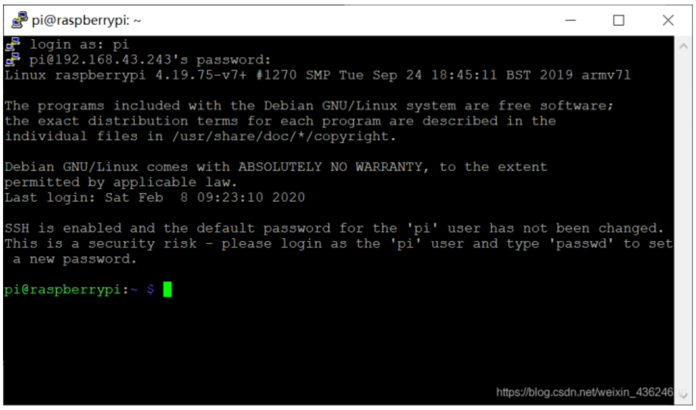


Image 11

10. Then, enter sudo apt-get install xrdp after successful login. (As shown in image 12)

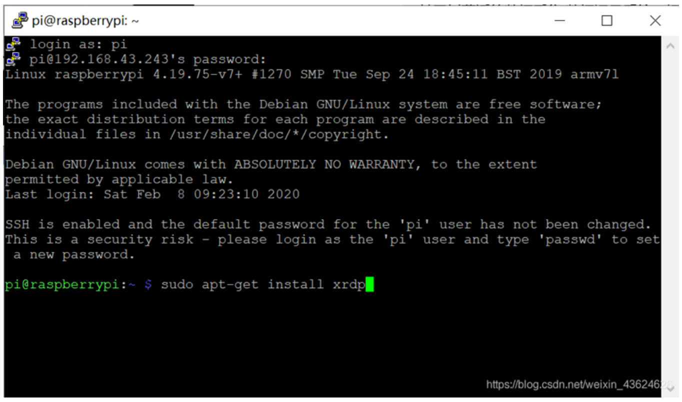


Image 12

11. Then, enter y, and press enter. (As shown in image 13)

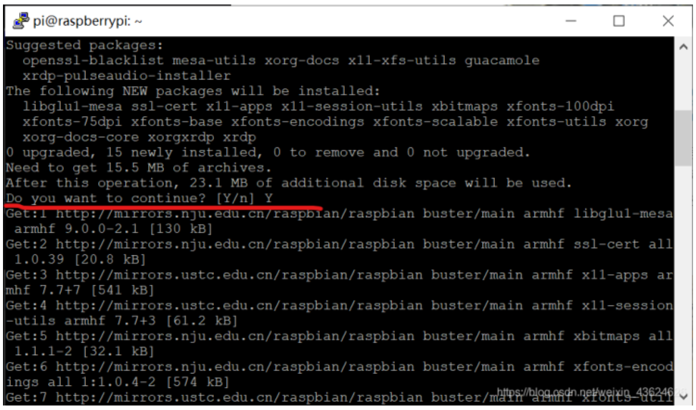


Image 13

12. Then open the remote desktop connection on your computer and enter raspberrypi username.
13. Press the connection and press yes in the new window.
14. Enter the raspberry pi's account and password (Account: pi & Password: raspberry) (As shown in image 14)

Class: EE486C
Team Number: 1
Project Name: Navigation Helper Helmet
Date: April 16th, 2021

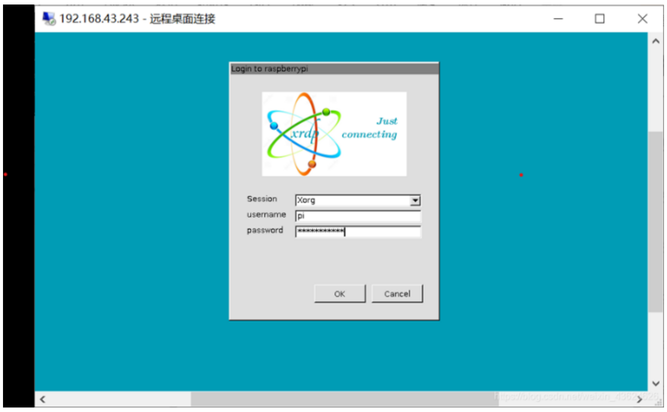


Image 14

15. Pass Ok to log in.

- Set boot script:

1. Download the helmet system of raspberry pi part, put the folder on the desktop.
2. You need to try to run all python scripts and use "pip install" to install all the packages you need to use.
3. Then Open the /etc/rc.local file. You can open it like image 15.

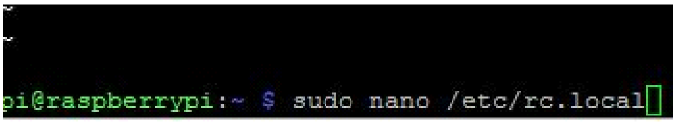


Image 15

4. Then type su pi -c "/home/pi/testboot.sh", before the exit 0 code. (As shown in image 16)

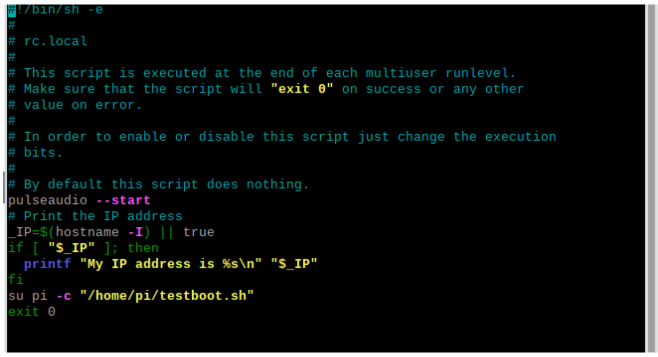


Image 16

5. Next, press ctrl+o to save, then exit the command window.
6. Then, create a new file and named it as testboot.sh, and write it as Image 17.

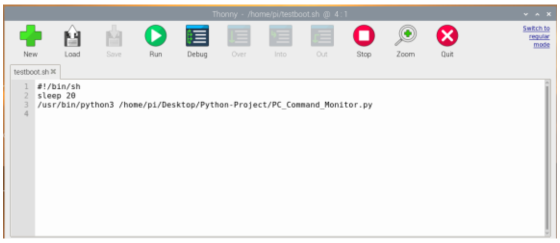


Image 17

7. Pay attention to file permissions. Access control should all be anyone. (As shown in image 18)

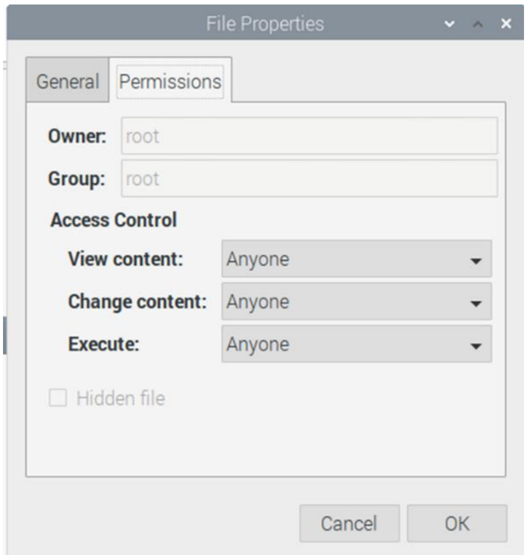


Image 18

8. Finally, save the file. Then exit and reboot the raspberry pi. Check the setting is successful or not by hearing the boot up voice command.

c. Configuration and Use:

When you have completed all the installation work according to the above content, you should copy all the programs to the Raspberry Pi and the computer according to the classification. Then comes the configuration and use part. ((As shown in image 19)

When you need to use our products, first you need to find and open our user interface. Our user interface does not mean a file with our team logo printed. Its name is "Navigation Helper helmet." It would help if you double-clicked to open it, and you will see the user login interface. In this interface, you need to enter the correct username and password to be able to log in to the real user interface (A complete schematic is shown in appendix 7). You can fill in the username and password you set at the time (the default username is pi, and the password is 12345678). So, you are in the user interface. Please note that you need to turn on the Raspberry Pi and wear headphones before proceeding. When you hear "the system has started" in the headset, you can start the next operation.

After you successfully enter the user interface, you will be able to see the function buttons on the left side of the user interface and explanatory text and prompts on the right side (A complete schematic is shown in appendix 8). Since our user interface has many functions, you need to understand the function of each button. First, you need to click "Parameter" to set the LIDAR monitoring range. Please note that since the unit of the monitoring range is meter, the only parameters you can enter are 1, 2, 3. Then please select the motor speed you need and click "Ok" to enter the next step when you have completed the selection.

Then, you can proceed with the following operations according to your needs. Click the

"Retraining" button when you want to add another person's face sample. Then you will see a pop-up window. You need to fill in the name of the object first and then fill in the number of samples you need to collect. Note that the number of samples can only be integers. Although there is no stipulation on the number of samples required, please collect as many samples as possible (generally 1000 photos are collected). If you need to use the recognition system, please first click the "Recognition" button, and wait a few seconds, then click the "Activate" button. Note, please be sure to follow this order. Because if you do not do this, it will cause a severe delay in the next video. When you follow the order, you will see the real-time video taken by the camera on the screen, and you can also hear the description of the voice system from the headset when the device detects the object. At this time, you can use the "o" key on the keyboard to turn on the system's video recording function or use the "p" key to turn off the system's video recording function. When the recording function is turned on, "REC" will be displayed at the top right of the screen, and when the recording function is turned off, "REC" will also disappear. If you need to turn on the motor rotation function, please press the "Motor" button. Please wait patiently for a few seconds after pressing, because the motor will only start to rotate after the voice playback is over. When you want to stop the motor, you need to click the "Motor" button again and wait for the motor to stop. You will hear "Motor reset" when the motor stops.

After you finish using the device, you don't need to do any debugging. Click "deactivate" to turn off the device completely. This is the voice you will hear "The system has been turned off." You also don't need to worry about the motor orientation because the motor will automatically recover every time you start the device. When the system is stuck, or the user does not operate properly, please click the "Reboot" button to restart the system. When you click the button, you will hear, "The system has restarted."

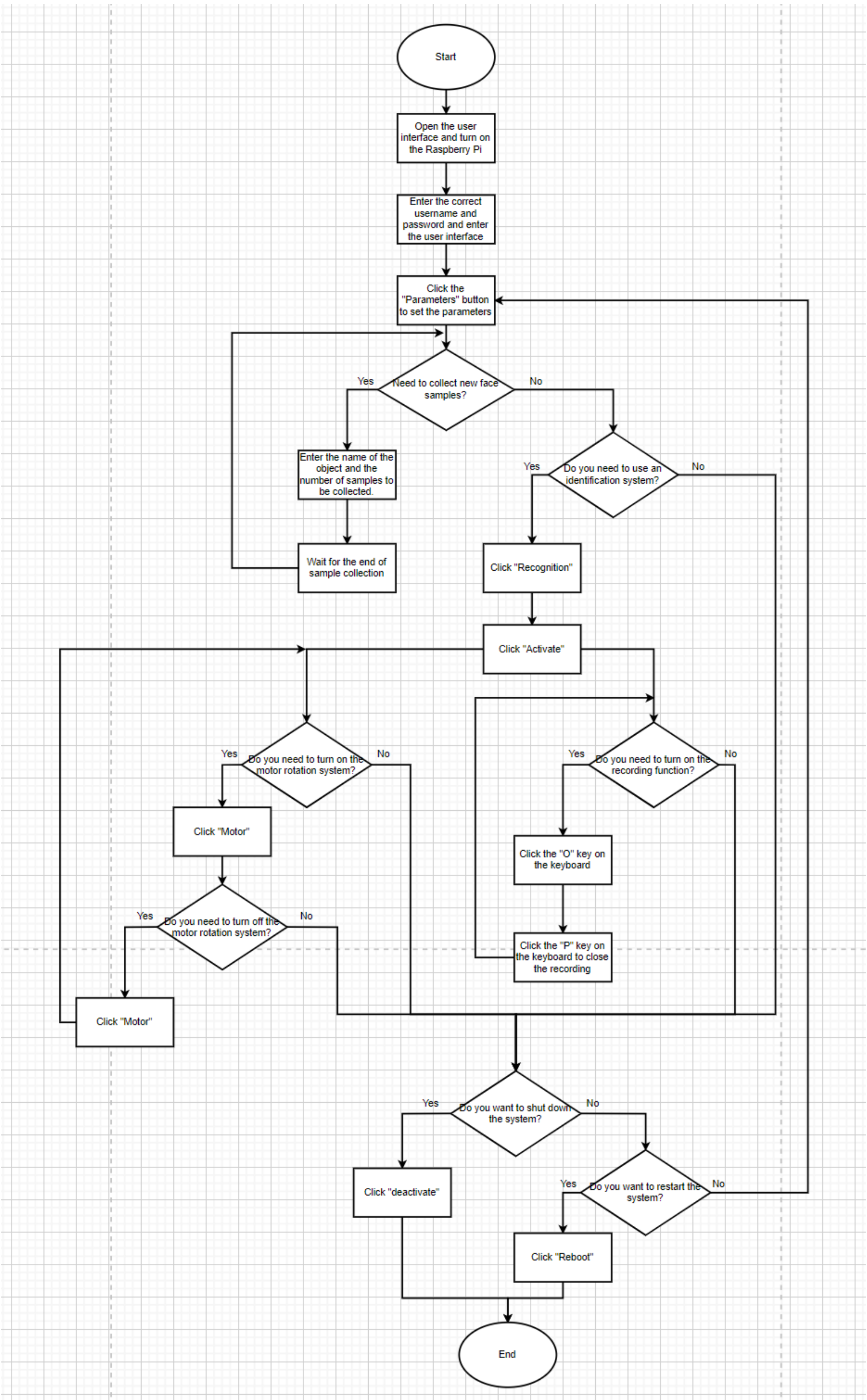


Image 19: Flow Chart of Configuration and Use

d. Maintenance:

Whenever the hardware and software have problems, they need to be repaired and replaced. This is divided into two parts: hardware part and software part. Details are as follows:

Hardware:

Please note that you must use the same products as the following hardware to replace to ensure the integrity of the system.

1. Raspberry Pi
   Model: Raspberry Pi 4 4GB
   Price: $55.00
   Purchasing URL:
   https://www.canakit.com/raspberry-pi-4-4gb.html
2. Camera
   Model: Raspberry Pi Camera Module V2-8 Megapixel,1080p (RPI-CAM-V2)
   Price: $24.45
   Purchasing URL:
   https://www.amazon.com/Raspberry-Pi-Camera-Module-Megapixel/dp/B01ER2SKFS/ref=asc_df_B01ER2SKFS/?tag=hyprod-20&linkCode=df0&hvadid=309776868400&hvpos=&hvnetw=g&hvrand=9770446950211685543&hvpone=&hvptwo=&hvqmt=&hvdev=c&hvdvcmdl=&hvlocint=&hvlocphy=9030289&hvtargid=pla-406302832745&psc=1
3. LIDAR Sensor
   Model: MakerFocus TFmini-s Micro Lidar Module
   Price: $39.09
   Purchasing URL:
   https://www.amazon.com/MakerFocus-Single-Point-Ranging-Pixhawk-Compatible/dp/B075V5TZRY/ref=asc_df_B075V5TZRY/?tag=hyprod-20&linkCode=df0&hvadid=241965663546&hvpos=&hvnetw=g&hvrand=12822846997480722226&hvpone=&hvptwo=&hvqmt=&hvdev=c&hvdvcmdl=&hvlocint=&hvlocphy=1013406&hvtargid=pla-454328524075&psc=1
4. Motor and Motor Driver
   Model: 28BYJ-48 DC 5V Stepper Motor and ULN2003 Driver
   Price: $8.99 for two
   Purchasing URL:
   https://www.amazon.com/KOOKYE-28BYJ-48-Stepper-ULN2003-Arduino/dp/B019TOJRC4/ref=asc_df_B019TOJRC4/?tag=hyprod-20&linkCode=df0&hvadid=198056450525&hvpos=&hvnetw=g&hvrand=6050843560311330294&hvpone=&hvptwo=&hvqmt=&hvdev=c&hvdvcmdl=&hvlocint=&hvlocphy=9030289&hvtargid=pla-384041320963&psc=1

Class: EE486C
Team Number: 1
Project Name: Navigation Helper Helmet
Date: April 16th, 2021
Software:

Please do not update the version of Python, and do not update all libraries in use.

1. Anaconda
   Version: Python 3.7 Version
   Download URL:
   https://www.anaconda.com/distribution/
2. cuDNN:
   Version: cuDNN7.4.1.5

   Download URL:
   https://developer.nvidia.com/cudnn
3. CUDA
   Version: CUDA10.0
   Download URL:
   https://developer.nvidia.com/cuda-10.0-download-archive?target_os=Windows&target_arch=x86_64&target_version=10&target_type=exelocal
4. Raspbian system
   Version: Depends on your computer's operating system
   Download URL:
   https://www.raspberrypi.org/downloads/
5. Putty
   Version: Putty 64Bit
   Download URL:
   https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html

e. Troubleshooting Operation

Although you have installed and debugged the device in accordance with the above method, you may encounter many problems in subsequent use. Therefore, to maximize your experience, we will list some possible failures and troubleshooting operations below.

- LIDAR is not working properly.
  1. First, compare the connection of LIDAR according to the installation guide. If there is no connection error, proceed to the next step.
  2. Please run the LIDAR script separately and check if there is any output. f there is no output, please replace the LIDAR sensor. If there is a correct output, proceed to the next step.
  3. Please check the distance.txt file. Check whether there are altogether blank characters, such as "space". Because the blank characters in the file will cause the speech system to read errors, which will cause LIDAR to not work properly. If there are blank characters, please delete them and fill in a number at will. If everything is normal and there are no blank characters, please proceed to the next step.

4. Please check whether the file path of the distance description in the voice system is correct. If it is not correct, please correct it to the correct file path.

- The camera is not working properly.
  1. Please check if the connection between the camera and the Raspberry Pi is secure. If it is stable, proceed to the next step.
  2. Please test the camera separately using the camera script. If it cannot be used normally, please proceed to the next step. If the camera can be used normally, please proceed to the next step.
  3. Please check whether the script of the remote recording is set correctly.
- The motor cannot rotate normally.
  1. Please follow the installation guide to check that the motor is installed as required. If there is no problem with the installation, please proceed to the next step.
  2. Please run the motor script separately. If the motor cannot be used, please try to replace the motor. If it still does not work, replace the motor driver. If it can be used normally, please proceed to the next step.
  3. Please check the delay time set by the motor, the delay number cannot be less than 2. If there is no problem with the delay setting, please proceed to the next step.
  4. Check whether there are blank characters in the degree.txt file. If yes, please change it to "0", and then manually turn the motor to the initial position. Test again.
- Voice cannot be played normally.
  1. Please check the path of the voice file. If the path is normal, proceed to the next step.
  2. If you are using a Bluetooth headset, please use the remote desktop to log in to the operation interface of the Raspberry Pi. Check whether the headset is connected to the Raspberry Pi via Bluetooth and whether the headset is set as the default audio output.
- Unable to open the user interface.
  1. Please check if you have downloaded and installed all required libraries and modules. If the installation has been performed correctly, please proceed to the next step.
  2. Check whether the versions of the libraries and modules are correct and modify them.

f. Conclusion

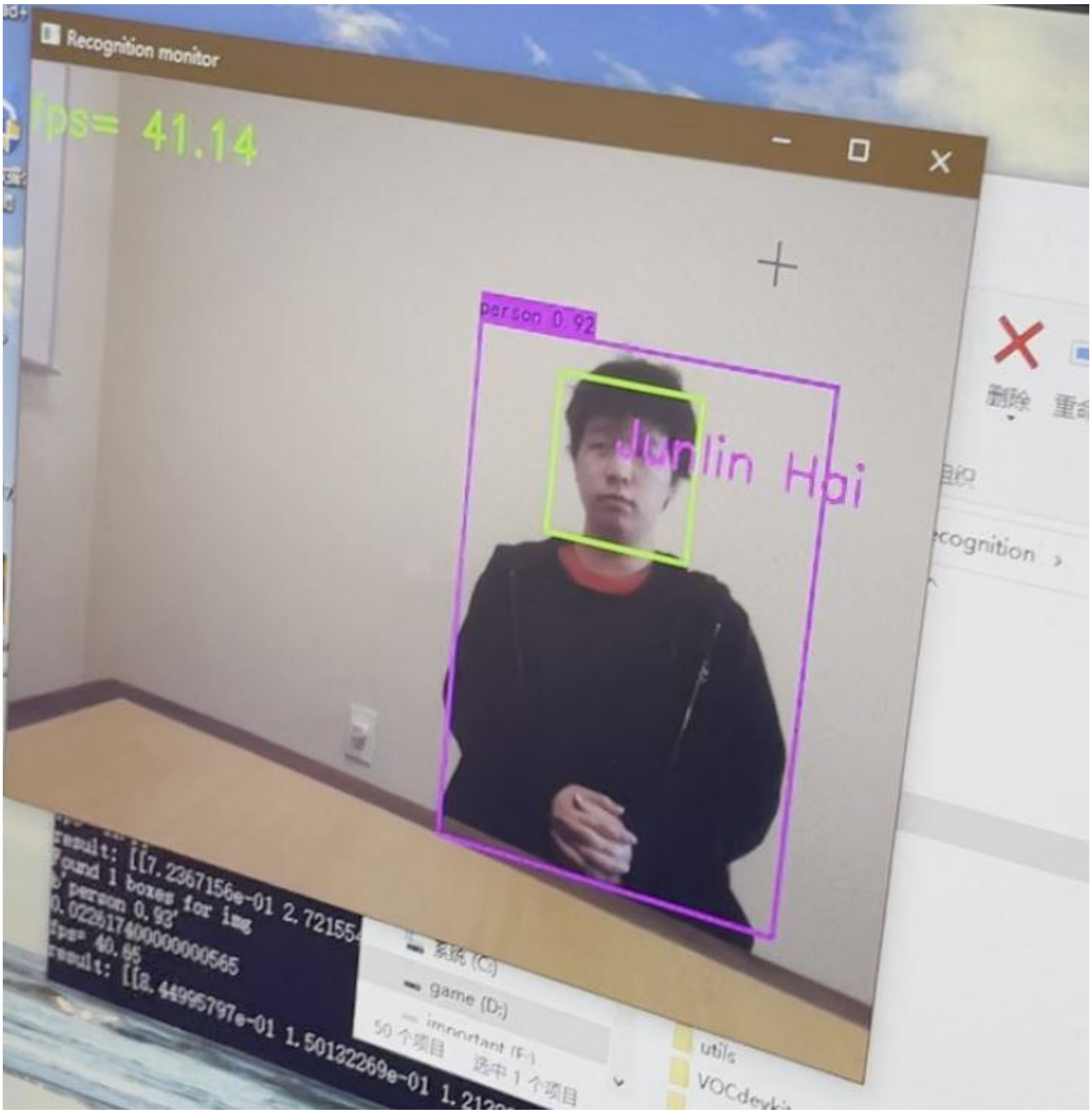
We are honored to be able to design this project with you. Thank you very much for your teaching and support to our work during this year. This year, we learned a lot of knowledge and accepted many challenges, but we did our best under your guidance. Although our work is over, we still hope to contribute to you and this project in the future. Therefore, if you have any questions, please feel free to contact us. We will try our best to help you solve these problems. I hope this can help you and your project.

Finally, I sincerely thank you. I wish you a happy use of our products.

Class: EE486C
Team Number: 1
Project Name: Navigation Helper Helmet
Date: April 16th, 2021

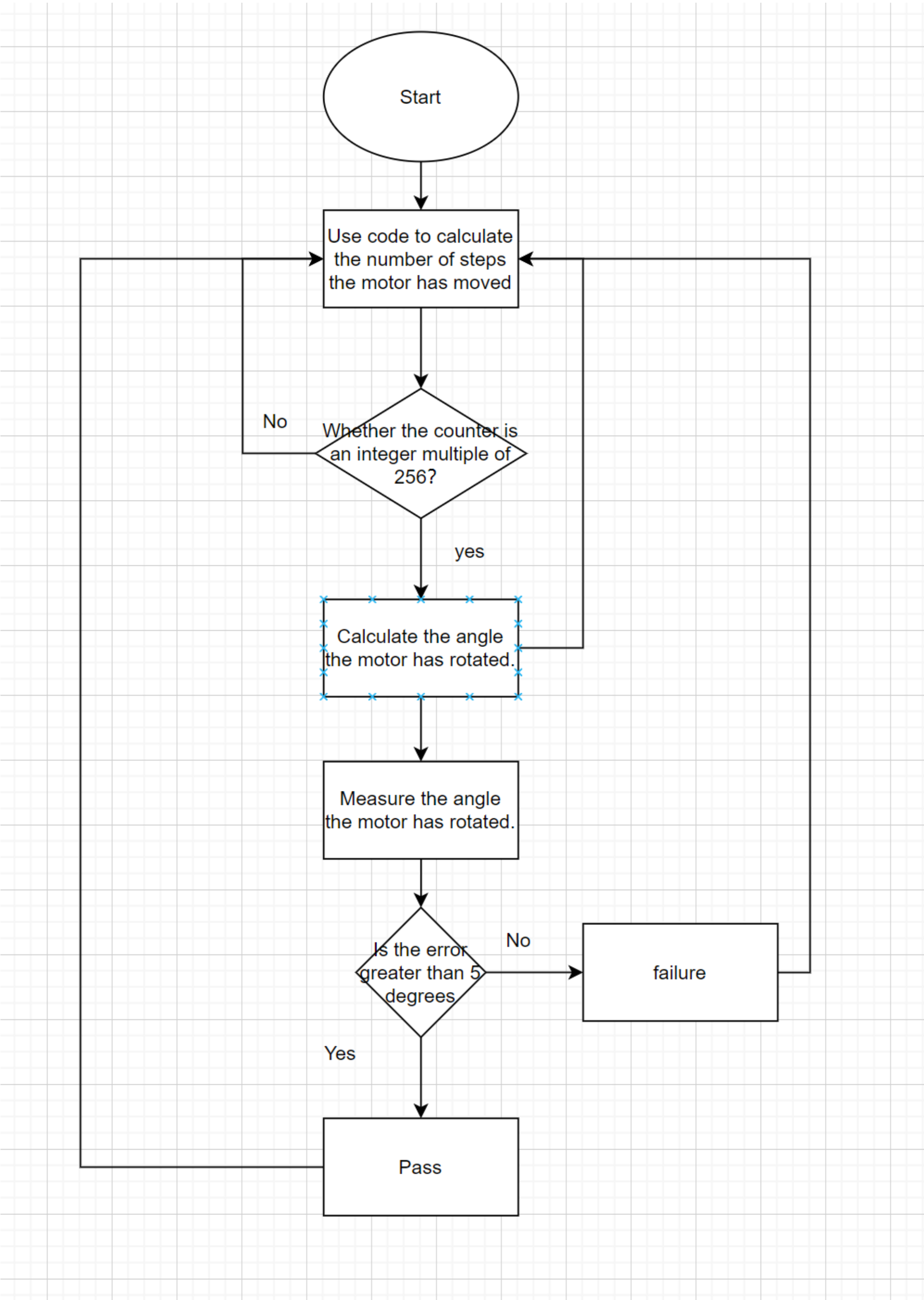
Appendices:

When performing face recognition, two boxes are processed, the larger one is the result of object recognition, and the smaller one is the result of face recognition.



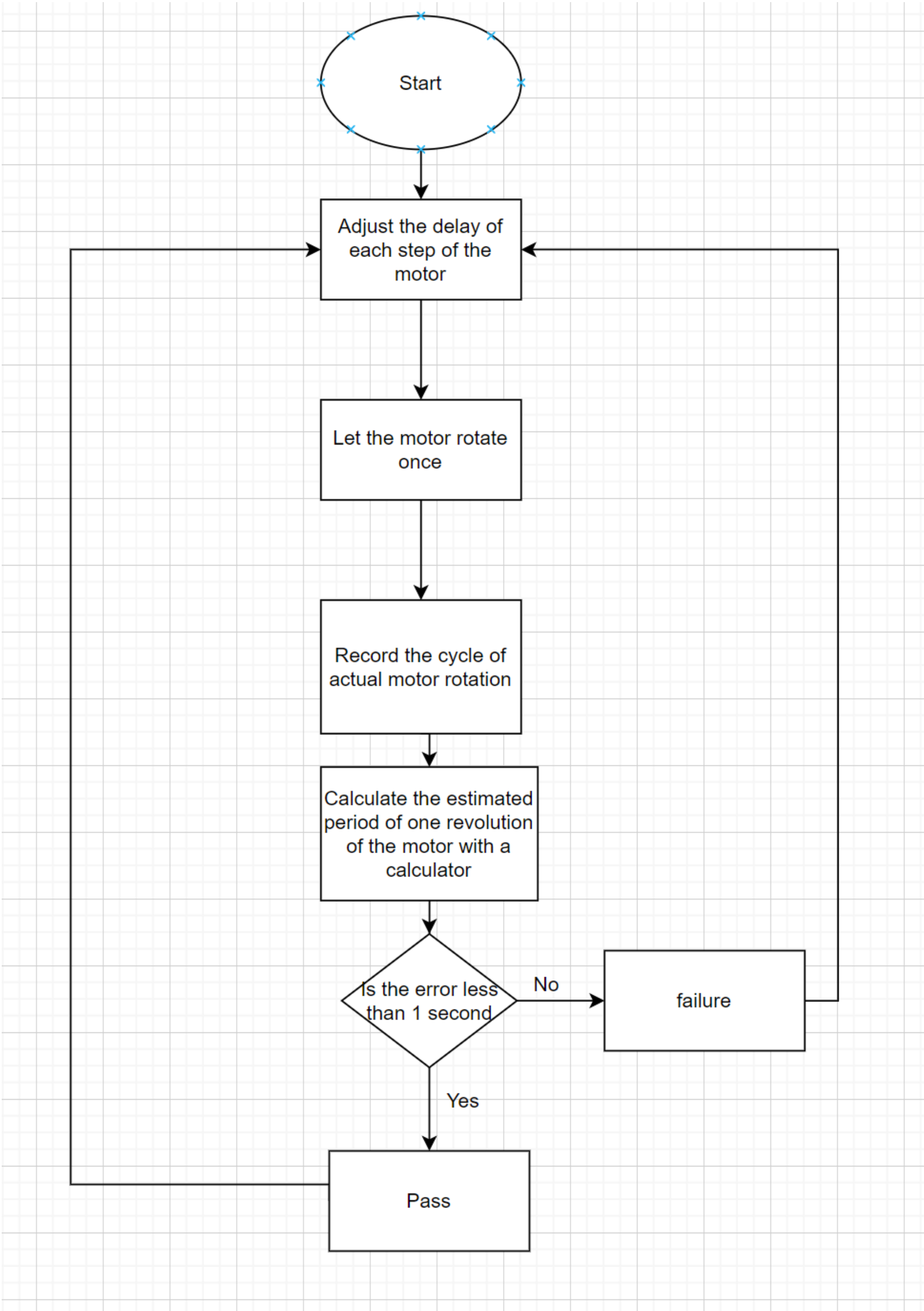
Appendix 1

Class: EE486C
Team Number: 1
Project Name: Navigation Helper Helmet
Date: April 16th, 2021
This is the flow chart of test 1.



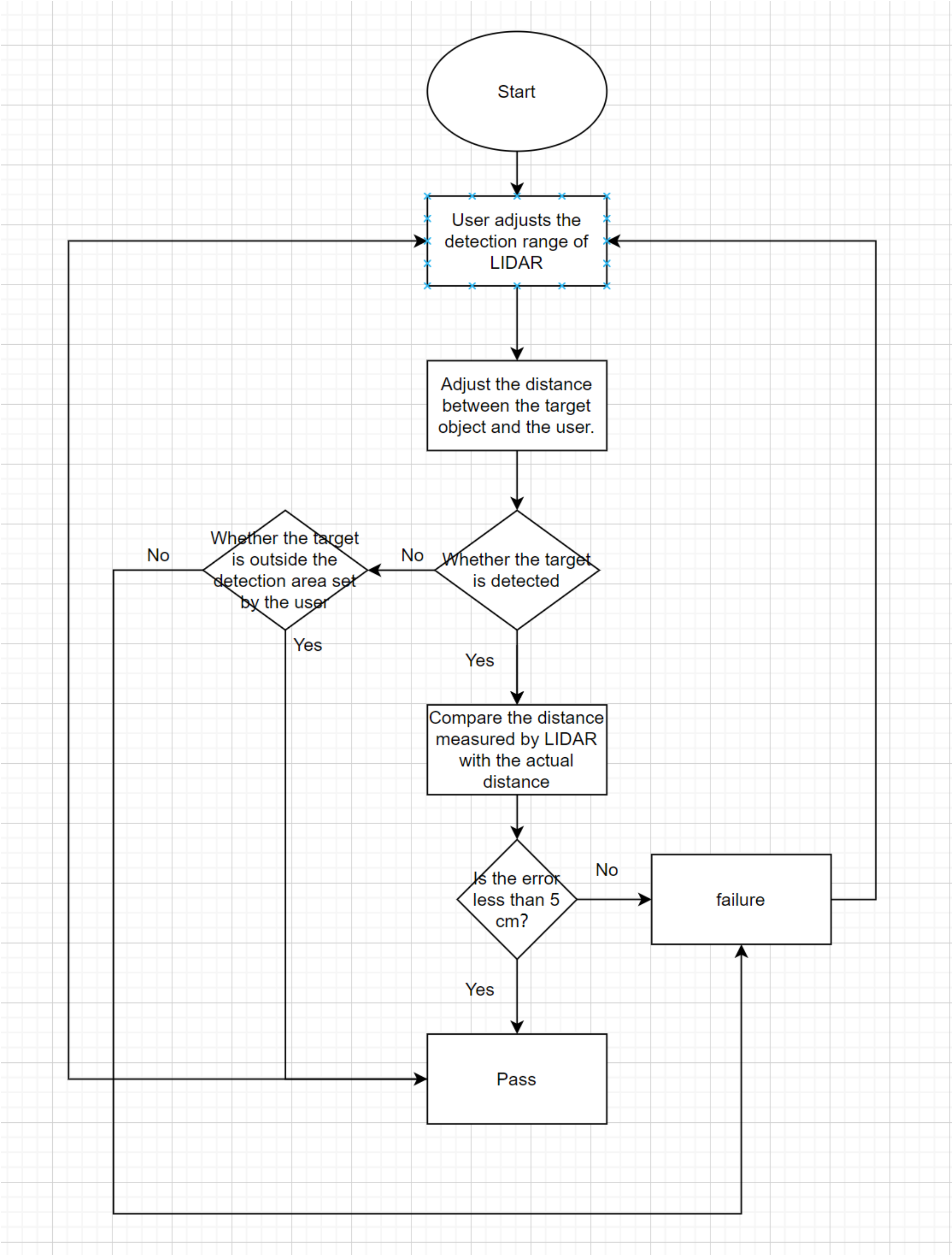
Appendix 2

Class: EE486C
Team Number: 1
Project Name: Navigation Helper Helmet
Date: April 16th, 2021
This is the flow chart of test 2.



Appendix 3

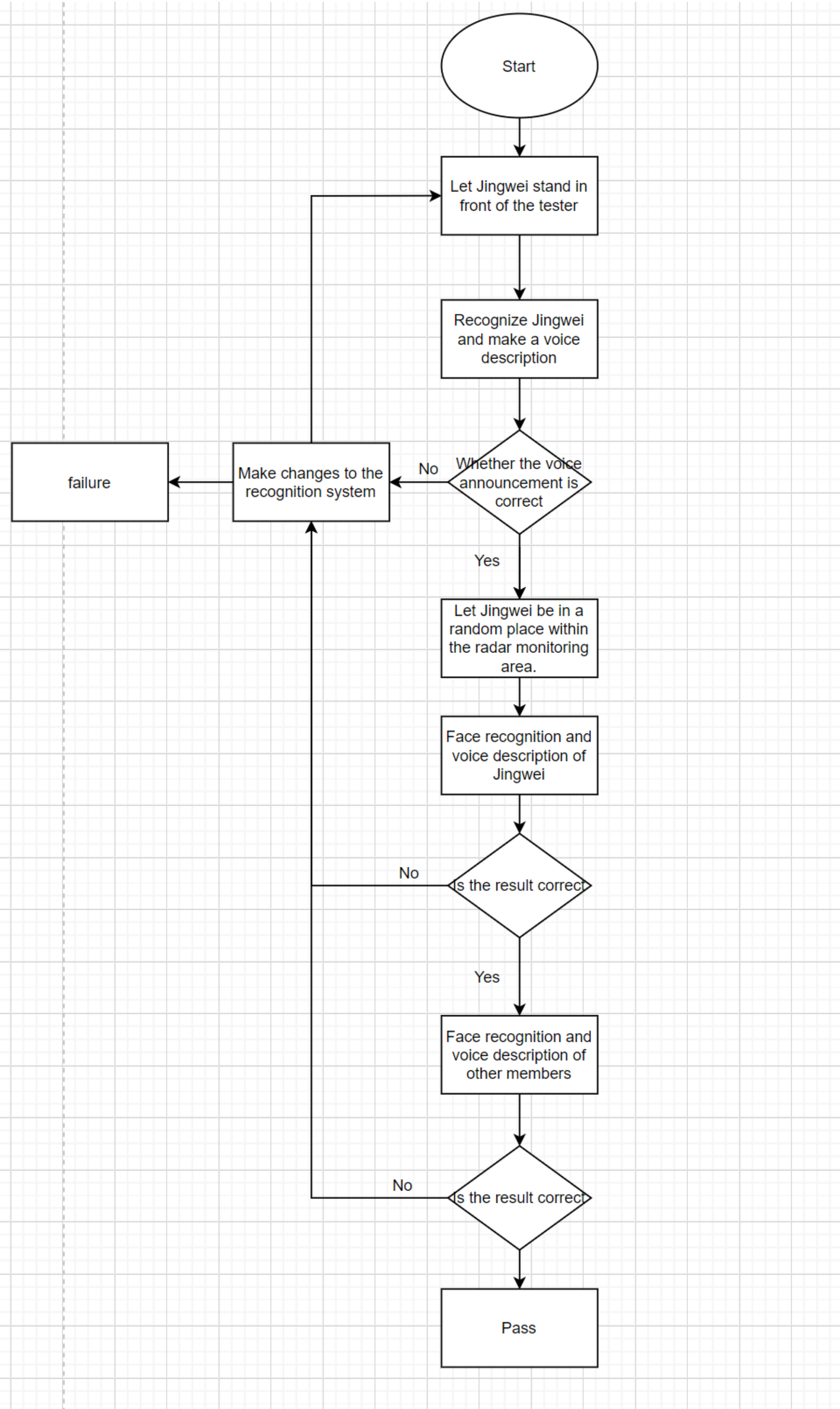
Class: EE486C
Team Number: 1
Project Name: Navigation Helper Helmet
Date: April 16th, 2021
This is the flow chart of test 3.



Appendix 4

Class: EE486C
Team Number: 1
Project Name: Navigation Helper Helmet
Date: April 16th, 2021

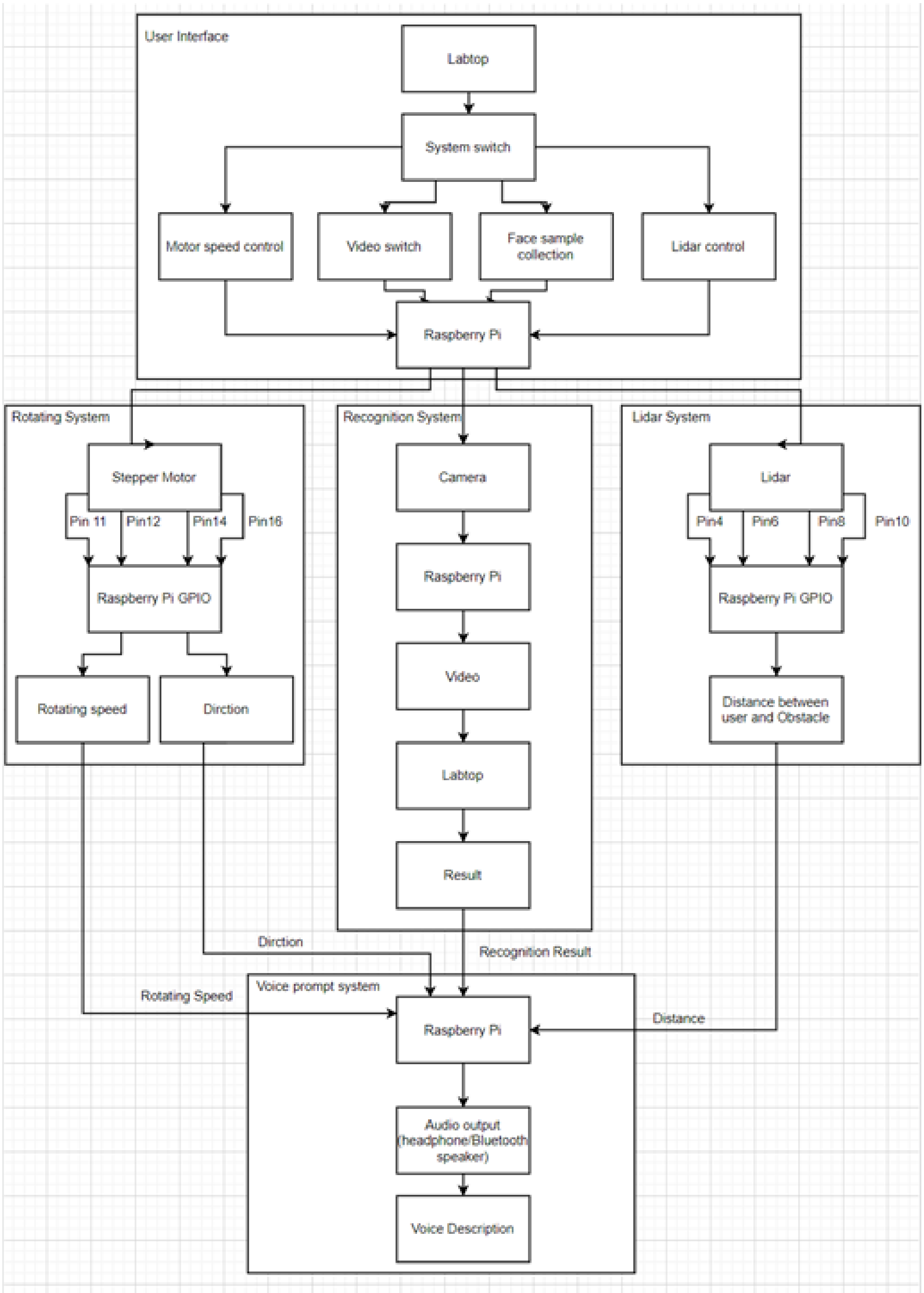
This is the flow chart of test 4.



Appendix 5

Class: EE486C
Team Number: 1
Project Name: Navigation Helper Helmet
Date: April 16th, 2021

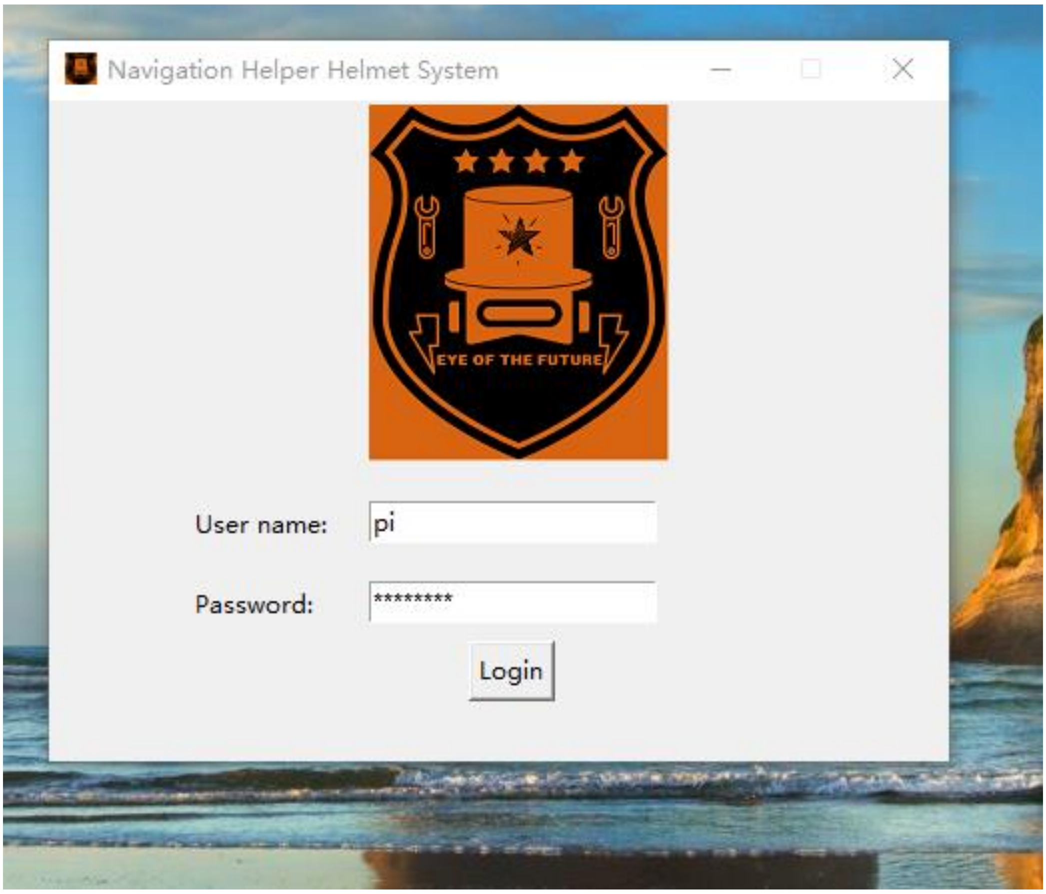
This is the System Architecture of our system. He noted the connection method of the LIDAR sensor and the motor, as well as the data exchange method.



Appendix 6

Class: EE486C
Team Number: 1
Project Name: Navigation Helper Helmet
Date: April 16th, 2021

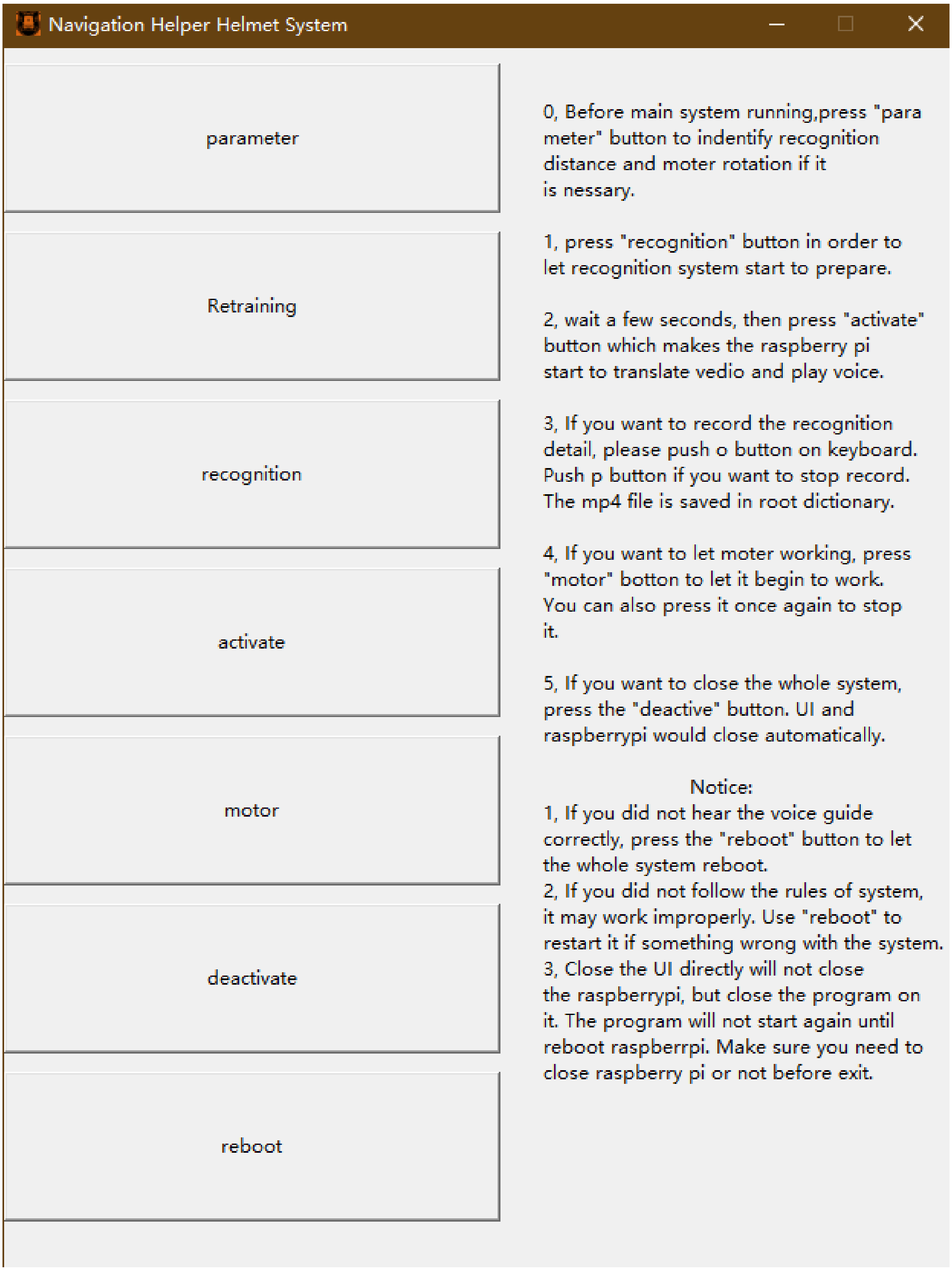
This is the login interface of the user interface. The user needs to enter the correct user name and password to log in.



Appendix 7

Class: EE486C
Team Number: 1
Project Name: Navigation Helper Helmet
Date: April 16th, 2021

This is the operation interface of the user interface. The 7 function keys on the left are instructions and precautions on the right.



Appendix 8