Robotic Hands Controlling By Smartphone App

Supervisors:
Dr. T. W. Chim
David Lee

Project Student:
Bert, Liu Kin Ning
Project Objective

During the inception phase of the project, certain ideas and concepts surrounding how to approach the tasks of creating the mobile application and robotic hands were formed. But the three months of learning, studying, programming and reevaluating have rendered many of the previous preconceptions not suitable for the whole project. As a result, a shift of focus and a change of project specifics have been deemed necessary in order to create something truly valuable in this final year project.

One major difference between the current mindset of the writer and what he had in september is that the former has paid a much higher attention to the software engineering side of the robotic hand, which comprises of, mobile application development, synchronization, ESP32 programming, the use of machine learning, the programming of sensors, instead of what he previously mainly focused on, which is the design of the shape of the robotic hand such that it mimics the shape of the human hand. The change of approach has been resulted from a number of reasons. First, the software aspect is much easier to handle than the design aspect. Second, by focusing on the making the softwares superior, the writer can implement a more thorough implementations of the softwares so that the whole system of the robotic hands and the mobile applications are more interesting. Thirdly, certain interesting ideas that emerged in the course of the three months, which the writer deem worth being added in the scope of the project, are preferred and which cannot possibly be added if the majority of the time is spent on perfecting the design of the robotic hand.
In short, there is a significant amount of change that the writer will include in the project and demonstrate in this report. A really short summary of the amended objectives of the project, however, is provided here in point form.

Objectives

1) To create a robotic hand that is flexible and can mimic many motions that the human hand is capable of.

2) Via delicate programming and software design, ensuring the movement and control of the robotic hand is smooth and not rigid.

3) To implement synchronization between the robotic hand and the mobile application such that the control is smooth and there is no issues concerning buffer overflow.

4) To implement sensors on the robotic hands such that it can not only be controlled but can receive signals via touching with the materials attached to the surface of the hand.

5) To implement a small-scale machine learning model on the mobile application so that it can use the camera to recognise hand gestures, then it can control the robotic hand based on what gestures the users make.

6) (Tentative) To create a set of APIs to control the robotic hand.

7) (Tentative) To allow the use of online platforms to control the robotic hand remotely via the Internet.
Project Background

As the part of the project background largely overlaps with what has been said in the previous initial report and will be further expanded in the final report, this part is not elaborated here.

Project methodology

The project is implemented with the use of 3d printing that creates the shape of the robotic hands, which is attached to strings connected to the motors that are connected to the motherboard. The motherboard itself is bluetooth enabled so that it can receive signals from a bluetooth device that is connected to it. The movement of the robotic hand is similar to that of the common non-motor based robotic hands controlled by humans.

The mobile application is created with React-Native, a mobile application development platform that allows developers to create apps that run on both iOS and Android devices with the minimal change of code and a single codebase written in Javascript. Database technologies are needed for the app to store states that are still persistent even after turning of the mobile application. To achieve this, WatermelonDB is used. To achieve a higher performance, the final product will be
programmed with ESP32, which has pins dedicated for sensors and has much better performance than Arduino Uno. The model of the robotic hands will be created using Sketchup Make.

**What has been accomplished**

By the time I write this interim report, the majority of the software parts of the project have already been accomplished. By this I mean the mobile application now already has a dedicate GUI and can perform the basic functions required. The board will be Indeed, most bluetooth-related communications have been programmed on both sides such that the motors connected to the motherboard can already be remotely controlled by the use of the mobile application via Bluetooth.

There are certain shortcomings currently. First, there are certain issues concerning the synchronization between the signal sent from the mobile application and received from the motherboard such that the buffer used to store bluetooth data does not have a good mechanism to ensure the communication is most smooth. Secondly, to quicken the process of implementation, the writer has been programming the motherboard with Arduino Uno, which has its limitations for memory and processing power and may suffer from performance issues if more features are added.

The basic features that were implemented in the first semester are introduced here.

**Bluetooth Connection**

As the mobile application communicates to the device via Bluetooth, the most important function is Bluetooth. The following pages are the appearance of the mobile application in the current stage. It should be noted that the GUI interface is still preliminary and improvements are due in the second semester.
This is the beginning of the application. As you can see, it encourages the user to connect.

This is the page that shows when you have to wait for the app to search for bluetooth devices.
This is the page that lists out the devices that can be connected, it shows the MAC addresses and names of the devices.

This is the page that shows when you are connected to a device
History feature

This is the page where all the previous history records are shown.

Pre-defined Module

This is the page that shows the gestures you can allow the hand to have
Recording function

This is the page where you can control the robotic hand and record a duration of the controlling so that it can later be replayed to replicate movements before. This greatly extends the possibilities of the application.
What will be done

A number of improvements to the existing problems, including the limitations of Arduino Uno and the synchronization issues will be solved. The former will be solved via the adoption of ESP32, which is smaller and more powerful. The latter will be solved by using a protocol that synchronises the movement between the hand and the messages sent to the hand. Other improvements such as those related to the GUI and UX will also be introduced as well.

Apart from the improvements, there is a list of things that will be done in the second semester as well. They serve to improve the functionality of the project and make it more interesting.

First off, capacitive sensors will be added so there will be sensory feedbacks going back from the robotic hand to the motherboard, so that better hand gestures can be performed because the robotic hand can move more naturally with the touch sensors.

Secondly, a tensorflow model of different hand gestures will be made so that we can use the camera to detect what gestures the users are making and the robotic hands can mimic the gestures. This can drastically improve the user experience.

Thirdly, the 3d modelling and printing of the robotic hand will be finished as well so that it can contain all the components and assemble the functionality of the robotic hand.

If time permits, A dedicated list of APIS for the controlling of the robotic hand will be included. Controlling via the internet will also be included as well to enhance the flexibility of the usage.