

Individual Project Assignment for COMP4336/9336 Mobile Data Networking Semester 2, 2018

Due: 11:59pm Friday 12 October

Weighting: 20% [20 marks]

Version 030918 – Released on 03 September 2018

Title: Mobile Gesture Communications using Light Sensor

Background and Motivation

With advancements in Internet of Things (IoT), we need to interact with many objects around us. Conventional methods for interacting with mobile devices, such as touch screen, verbal commands, etc. will not always be feasible, convenient or privacy-preserving. Instead, *hand gesture* is expected to become the preferred mode for communicating with many IoT devices. Researchers are looking to exploit various sensors such as camera, accelerometer, microphone, and light sensor to recognize gestures. Tech giants such as Microsoft Kinect and Leap Motion have already implemented some form of gesture recognition in their products. In this assignment, you are required to design a mobile gesture communication system for Android devices using light sensors.

Learning objectives

Upon completing this assignment, students will:

1. Master the access and manipulation of light sensor in mobile devices,
2. Design and implement a real-time gesture recognition system for mobile devices, and
3. Develop an application that takes input from hand gestures.

Assignment Tasks

Task 1 Graphical Display of Light Sensor Data [3 marks]

A light sensor measures the *light intensity* of the environment. The brighter the environment, the higher the sensor readings, and vice versa. The key idea of light-based gesture recognition is that specific hand movements (gestures) near the device block ambient light in specific ways, leaving their unique signatures on the light sensor readings. Hence, the first step is to obtain the *time series* of light sensor data from your Android device. In this task, you should develop an App that can continuously read light sensor data and display a real-time graph of the sensor values (only display the latest five seconds values).

Task 2 Gesture Counter [3 marks]

In this task, you should define a *basic gesture*, **Down-Up**, which represents moving a hand towards the light sensor and then moving away from it (assuming the device is lying on a table). In the real-time graph, you can expect a *trough* as the sensor value would decrease when the hand is

approaching the device, and vice versa. Your aim in this task is to *count* the number of *troughs* within the last five seconds and display the *trough counter* in a *TextView*. To achieve this, you will have to design an algorithm to detect *troughs* in light sensor time series data.

Task 3 Gesture Recognition [5 marks]

By exploiting the *counts*, as detected in Task 2, you can define a set of gestures with *different counts*. For example, *one count* (only one trough detected) represents *Gesture1*, *two counts* (two troughs detected) represents *Gesture2*, and so on. In this task, you are required to develop a practical App that can recognize count-based hand gestures and react to them. Examples of *practical* apps can be a **gesture-controlled music player** that allows the user to control the functions of the player, such as *play*, *pause*, *turn up the volume*, *turn down the volume*, etc. using hand gestures. Another example app can be a **gesture-assisted photo sharing via Bluetooth**. In Lab7, you will learn Bluetooth based device-to-device communication. However, all the actions in that lab experiment are based on your typing inputs (e.g., click buttons). In this example app, you are expected to share a photo with a nearby Bluetooth receiver (another smartphone or laptop) by relying only on your gesture inputs (without typing). You are allowed to design any other *similar* apps (should be a *practical* app) that exploit count-based gestures. Your apps should recognize at least 3 gestures.

After you implement your application, you should measure its recognition performance under different lighting environments (e.g., indoor vs. outdoor). In terms of the performance, you should report two metrics:

Miss rate: the percentage of gestures miss detected.

Recognition accuracy: the percentage of the correctly recognized gestures among all the detected gestures.

For example, if you performed 10 gestures, 8 gestures are detected in which 6 are correctly recognized (i.e., your app correctly reacts to the gestures), the two metrics should be: miss rate = $(10-8)/10 = 20\%$, recognition accuracy = $6/8 = 75\%$.

Some Useful Links

Gesture recognition

<https://www.cs.dartmouth.edu/~trdata/reports/TR2016-797.pdf>

Android music player (if you are developing a gesture-controlled music player)

<https://developer.android.com/guide/topics/media-apps/audio-app/building-an-audio-app>

Bluetooth file transfer (if you are developing gesture-assisted device-to-device photo sharing)

<https://www.intorobotics.com/how-to-develop-simple-bluetooth-android-application-to-control-a-robot-remote/>

<https://developer.android.com/guide/topics/connectivity/bluetooth>

<https://dzone.com/articles/bluetooth-data-transfer>

Moodle Discussion Forums

You can discuss any issues with the assignment specs in Moodle Forum.

Assignment Submission and Marking

Assignment will be marked based on a report (8 marks) as well as a demo (12 marks) that you will have to perform during Week 13 (demo slots will be advertised later). You are required to submit your code and the report by 11:59pm 12 October. Late penalty at the rate of 10% per day late will apply for the report submission. Submissions will be accepted after 19 October.

Demo [12 marks = 3 (Task1) + 3 (Task2) + 5 (Task3) + 1 (Questions)]

Each student will be allocated 15 minutes for demo on Week 13. During the demo, you should download the code you submitted by 12 October and run it on an Android phone (personal laptop and phone are accepted). You can get corresponding marks for each task if the functionality is accomplished. You will also be asked questions about your design, e.g., (1) have you tried different counting algorithms and compare their performance? (2) what is the most difficult part in your implementation and why? Based on your answer, you can get the marks for questions.

Report Submission [8 marks]

Write a report including the following:

- Conduct a literature review on gesture recognition with at least two pages, in which you should (1) investigate different modalities (e.g., Wi-Fi, sound wave, light, and etc.) and algorithms used for gesture recognition; (2) compare their performance and analyze their strengths and limitations. **[2 marks]**
- Describe the algorithm you designed and implemented for counting the number of *Down-Up* gesture. **[Task2, 0.5 mark]**
- Which functionalities of your application are gesture-controlled and what are the corresponding gestures. **[Task3, 0.5 mark]**
- What is the performance of your application under different lighting conditions **[Task3, 2 marks]**
- What are the challenges you encountered and how you solve them **[1 mark]**.

The report must be submitted in PDF format using Turnitin in Moodle. The report must comply with the following format **[1 mark]**:

- Name of the file should be in the format of <first name>_<last name>.pdf
- The first page of the report must include the name and ID of the student
- The report must be limited to a maximum of 10 pages, 2 cm margins on all sides, 10-pt Times New Roman font, single spacing and single column.
- The size of the PDF file should be less than 10 MB

The remaining **[1 mark]** is for the grammar.

The End (We hope you enjoy doing this assignment)
