

**Human Activity Recognition   
through Wearable Devices**

**Edwin Candinegara**

**Supervisor:** Prof. Tan Ah Hwee

**Examiner:**

**March 2017**

School of Computer Science and Engineering



**SCE16-0405**

**Human Activity Recognition   
through Wearable Devices**

Submitted in Partial Fulfillment of the Requirements for the Degree of Bachelor of Engineering (Computer Science) of the Nanyang Technological University

by

Edwin Candinegara

**March 2017**

School of Computer Science and Engineering

Table of Contents

[Abstract 3](#_Toc472005325)

[Acknowledgment 4](#_Toc472005326)

[1. Introduction 5](#_Toc472005327)

[1.1 Background and Motivation 5](#_Toc472005328)

[1.2 Project Objectives 6](#_Toc472005329)

[1.3 Scope and Limitations 6](#_Toc472005330)

[1.4 Report Overview 7](#_Toc472005331)

[2. Literature Review 9](#_Toc472005332)

[3. Analysis on Previous Work 11](#_Toc472005333)

[4. Frameworks and Tools 13](#_Toc472005334)

[4.1 AngularJS 13](#_Toc472005335)

[4.2 Tornado 13](#_Toc472005336)

[4.3 Scikit-learn, Pandas, and NumPy 13](#_Toc472005337)

[4.4 MongoDB 13](#_Toc472005338)

[5. Issues and Challenges 14](#_Toc472005339)

[a. Human Activity Recognition 14](#_Toc472005340)

[b. Real-time Recognition System 14](#_Toc472005341)

[6. Approaches and Technologies 15](#_Toc472005342)

[a. Tools 15](#_Toc472005343)

[b. Data Pre-processing and Prediction Pipeline 15](#_Toc472005344)

[7. Project Tasks 16](#_Toc472005345)

[a. Code Rewriting 16](#_Toc472005346)

[b. Building Data Pipeline 16](#_Toc472005347)

[c. Enhance the Android Application to Perform Real-time Recognition 17](#_Toc472005348)

[d. Data Collection and Model Building using Sensor Data from Smart-watch 17](#_Toc472005349)

[e. Model Testing and Improvement 17](#_Toc472005350)

[8. Schedules 18](#_Toc472005351)

[9. References 19](#_Toc472005352)

# 

# **Abstract**

# **Acknowledgment**

I would like to express my gratitude to Prof. Tan Ah Hwee and Dr. Wang Di for their direct supervision and guidance throughout my final year project. Their dedication and involvement in my final year project progress have made it possible for me to finish this final year project. Furthermore, the discussion during the weekly meetings conducted by Prof. Tan and Dr. Wang made sure that the final year project is on track and generated many new ideas for improving the work.

I would also like to thank my family for their continuous support throughout my whole study period here in NTU. Next, I would like to thank all of my friends for their support, especially for those who spent their time helping the data collection for this project, and the fun we had together.

Finally, I would like to extend my gratitude to Google scholar for the ease of accessing and finding related papers and Stack Overflow community for answering many programming problems faced during the course of this project.

# **Introduction**

## **Background and Motivation**

With increasing life expectancy and lower birth rates trends in Singapore, the number of citizens aged 65 and above has increased significantly in the last few years. In addition, this trend is expected to continue in the next few years and by 2030, the number of citizens aged 65 and above will be doubled [1]. Furthermore, according to a survey [2], it is shown that almost 90% of those aged 65 or above indicate that they want to stay at home as long as possible. Thus, there is an increasing demand of technologies which are capable of continuously monitoring elderlies without being intrusive to the users.

Human activity recognition technology is the key technology for monitoring elderlies in an effective manner. This research area has been studied by many for the last decade. Researchers have presented a number of different ways for recognizing human activities using different kind of technologies. Initially, researchers used specially made sensors which are expensive and need to be placed in certain positions [3, 4]. This setting is not suitable for conducting daily activities. Thus, in recent years, researchers are focusing more on using non-intrusive technologies to perform human activity recognition.

Video surveillance is one of the most commonly used technologies. Cameras are installed inside a house and the system is also capable of providing live video streaming to the caregivers. In recent years, with the advancement in artificial intelligence and machine learning, video surveillance systems are now able to detect the activities performed by the elderlies [5, 6]. With this capability, caregivers need not constantly check the live video. The system would be able to notify caregivers only when important events occur. However, video surveillance systems may cause privacy issue in case the video is accessed by an unauthorized party.

Smartphone is another commonly used technology for recognizing human activities. Smartphones are equipped with many sensors such as accelerometer, gyro meter, and barometer which can be used for recognizing human activities. The smartphone only needs to be put inside the user’s pants. One major drawback of only using smartphone for recognizing human activities is that it will not be able to accurately distinguish hand-based activities.

Beside smartphone, wearable devices are also increasingly prevalent. Smartwatch, for example, is an emerging wearable device and there are many kind of smartwatches commercially available [7]. A commercial smartwatch has multiple embedded sensors which are suitable for recognizing human activities. By using sensory data retrieved from both smartphone and smartwatch, one can now also recognize hand-based activities much more accurately. This will allow people to track what the elderlies are doing at home more accurately and can cover a wider range of different activities.

## **Project Objectives**

There are 3 main objectives this project would like to achieve:

* 1. **Build a new human activity recognition system using features extracted from sensory data gathered from both smartphone and smartwatch.** The system developed by the previous students in [8] and [9] was only using sensory data collected from a smartphone. In this project, sensory data collected from a smartwatch will also be incorporated to improve the prediction accuracy. Furthermore, more activities will be considered in this project, especially activities which involve hand movements.
  2. **Collect dataset from multiple test subjects.** The dataset will be mainly used for thorough experimentation and testing of the developed human activity recognition system. This dataset will be used for not only K-Fold Cross Validation, but also for Leave-One-Person-Out (LOPO) Cross Validation to ensure the system’s ability of accurately predicting activities conducted by a new user not in the dataset. Furthermore, with this dataset, the results of the model trained using data only from each device can be compared with the model trained using data from both devices combined.
  3. **Build a system which is capable of recognizing the motions of an individual with minimum delay.** This objective includes the development of an Android smartphone application and a Tizen smartwatch application for collecting the sensory data, a web application for showing the predicted activity, and a backend system for processing the collected sensory data and predicting the activity using the human recognition system developed.

## **Scope and Limitations**

The main scope of this project is the development of a human activity recognition system for detecting different physical activities of an individual. This involves the creation of an end-to-end data pipeline for model training and testing, including data sampling, data windowing, and features extraction. In addition, training and testing data will be taken from 15 test subjects within the age range of 18-24 years old and motion transitions are not covered in this project. Each test subject is asked to perform 14 activities as listed in Table 1. Finally, an ensemble machine learning classification algorithm called Random Forest [10] and the Support Vector Machine (SVM) algorithm will be used in this project.

The project also involves the development of a web application which is able to recognize an activity with minimum delay. As such, a system needs to be developed with a separate pipeline to process incoming sensory data sent from the smartphone and smartwatch periodically. Furthermore, delays are inevitable since only after one data window with a configurable number of seconds is obtained, then it can be sent for processing. Network latency is also another source of delay since data is sent to the backend server through the internet.

**Table 1.** The list of activities studied in this project.

|  |  |
| --- | --- |
| Basic Activities | Hand-based Activities |
| Standing | Writing |
| Sitting | Typing |
| Lying | Reading |
| Walking | Food Preparation |
| Running | Sweeping the Floor |
| Going Upstairs | Brushing |
| Going Downstairs | Folding a Shirt |

## **Report Overview**

This report is organized as follows:

1. **Section 1** introduces the background and objectives of this project.
2. **Section 2** provides the literature review of related work done in the field of human activity recognition technology as well as the work done by the two previous students.
3. **Section 3** provides an analysis over the work done by the two previous students.
4. **Section 4** introduces the frameworks, technologies, and tools used in this project.
5. **Section 5**
6. **Section 6**
7. **Section 7**
8. **Section 8**

# **Literature Review**

Human activity recognition is a technology for recognizing actions performed by a human by processing a series of observations. These observations are usually obtained using sensors found on smart devices such as smartphones and smartwatches. In the recent years, there have been a lot of studies conducted in this area. Researchers have studied various data collection, data pre-processing, and model building methods.

Many data collection methods have been investigated by researchers. Researchers in [8], [11], [12], [13], and [14], for example, used devices strictly placed at fixed positions and orientation on the test subject’s body. In [13] and [14], the data collection is done by using a specially made hardware with one or more sensors while in [8] and [12], a smartphone with multiple embedded sensors is used. The hardware or smartphone is placed in a fixed place around the test subject’s pelvic region or waist. As the same setup condition is imposed on all test subjects, the data variance is minimized and the model can perform much better.

In order to handle varying positions and orientations of the mobile phone, researchers in [15] introduced the usage of the accelerometer magnitude as a feature. They proved that the model can handle various smartphone positions and orientations. As a result, the authors of [15] achieved better results for human activity recognition with more natural data collection settings. The same technique was also used in [9] and [16]. The authors in [9] and [16] reported that the trained model can produce very accurate predictions using the Support Vector Machine (SVM) machine learning algorithm.

It is also observed that there are only a few activities considered in many past studies. These activities are low level motions such as sitting, lying, walking, jogging/running, and standing. These activities were considered in [9], [12], [16], and [17]. Furthermore, researchers in [11], [15], and [18] included additional motions such as going downstairs and upstairs. These two activities are usually confused with walking as they are very similar. Researches that only used specially made hardware or smartphones did not consider more complex activities (e.g. typing and writing) as these activities will be very hard to differentiate.

As the popularity of wearable devices is rising recently, researchers started to investigate the usage of these devices for data collection. Most studies in this area used smartwatches which are commercially available in the market as can be seen from [19], [20], [21], [22], and [23]. Smartwatch is quite a popular wearable device for this research area as it has multiple embedded sensors and is relatively cheap. On the other hand, researchers in [24] tried to mimic a smartwatch by placing a smartphone on the test subject’s wrist area while a specially made hardware was used in [25].

With the addition of a smartwatch, more complex activities can be included in the study, especially those involving hand movements. The sensors embedded in the smartwatch would be sufficient to capture the hand movements. For example, [23] and [24] included hand-based activities such as writing and typing. There are other hand-based activities considered by researchers but each research has its own different set of hand-based activities. Moreover, although the existence of smartwatch allows more complex activities, the work in [21] still only considered common lower level motions, i.e., sitting, standing, lying, walking, and running.

The results reported in [20] were considerably good for a model trained on a dataset with more than 20 different activities. There are motion transitions, lower level motions, and more complex motions involving hand movements. The algorithm used was the Support Vector Machine (SVM) with Pearson Universal Kernel (PUK). The model trained using only the smartwatch data with the top 30 features can achieve a mean F-score of 0.93. On the other hand, the best accuracy in [23] was only 70.3% which was achieved by a Random Forest model trained using the smartwatch accelerometer data. The results in [24] were also not as good as the results in [20]. These differences could be caused by the algorithms used, the extracted features, the types of activity, and the data collection methods.

As previously mentioned, study in the area of human activity recognition technology with wearable devices only attracts researchers recently. It can also be seen from previous studies that there is a room for improving the results. In this project, we aim to investigate whether the features used in [9] and [16] can achieve good results in classifying not only lower level motions, but also more complex hand-based activities. In addition, we would like to investigate the significance of other sensory data such as barometer and gyroscope sensor. Using barometer data, activities like going upstairs and going downstairs should be differentiable from walking and/or running.

# **Analysis on Previous Work**

This project is related and a continuation of the previous work done by previous students in [8] and [9]. As the work in [9] is also an improvement of the work in [8], the discussion in this section will mostly refer to the work done in [9]. The human activity recognition model introduced in [9] is only trained using 22 features extracted from a smartphone’s accelerometer data and the learning algorithm is the Support Vector Machine (SVM) with Radial Basis Function (RBF) as its kernel function. Furthermore, there are 5 lower level motions (standing, sitting, lying, walking, and running) and a number of higher level motion transitions. This project will extend the previous work in the sense that the data collection will also involve a smartwatch and more activities are considered.

The author conducted thorough testing on the developed SVM model using two well-known model validation techniques: K-Fold and Leave-One-Person-Out (LOPO) cross validation. In addition, the author reported how using a number of different window sizes affect the performance of the model. Generally, having a bigger window size will allow the model to derive more information and hence better prediction results. However, using bigger window size may not be suitable for real time human activity recognition use case.

One of the issues the work in [9] was the orientation and position of the smartphone inside the test subject’s pocket. It is common for the smartphone to slip and slightly tilt from the original position which may cause noisy sensory data. The author allowed test subjects to choose which front pocket to use (left or right) and the type of clothing worn during the data collection session. In order to handle this problem, the author adopted the idea of incorporating the magnitude of the accelerometer reading as a feature based on [15]. Moreover, with only 22 features, the model can achieve 99.67% mean F1 score for the 10-fold cross validation on all the collected dataset and 94.25% mean F1 score for the LOPO cross validation.

On the other hand, although the results seem very promising and the size of the dataset is considerably large, 8 hours of combined youths and elderlies data, it is important to note that the dataset was only collected from a small number of test subjects (4 youths and 4 elderlies). Each test subject may perform the 5 lower level motions in a different manner from other test subjects. In order to confirm that the model is robust and can generalize to unseen data from a new test subject, the model should be tested on a dataset collected from more test subjects to incorporate more variance in the data. For example, collecting 3 minutes of “walking” activity for each test subject from 10 different test subjects may allow us to observe more variance in the collected data compared to collecting 30 minutes of “walking” activity from the same person. As a result, the LOPO cross validation result of the developed model is considerably high probably because of the less variance in the collected data and hence, the model can draw the decision boundaries better.

The data collection conducted in [9] was only using a smartphone placed inside the test subject’s left or right front pocket. As previously discussed, using only a smartphone means the number of activities which can be accurately distinguished is limited to activities having different lower body motions or positions. For instance, distinguishing “writing” and “sitting” activity would be very hard if the data is only collected using smartphone since the only difference is the hand movements. With the addition of a smartwatch, more complex activities that involve hand movements can be considered and this is the main objective of this project.

Another observation from the work done in [9] is that there is no data pre-processing pipeline developed. This means that the raw data pre-processing steps had to be done manually by a human operator which is very tedious, very prone to human error, and may affect the correctness of the results due to the human errors. Thus, this project aims to build an automated the data pre-processing pipeline so that any raw data collected can be transformed into a format needed for the model training and testing just by running a single script. This will significantly reduce the human error factor in this project and quicken the development process as it is more organized and the data pre-processing steps are automated.

# **Frameworks and Tools**

There are a number of different programming languages, frameworks, technologies, and tools used in this project. There are three programming languages mainly used throughout this project: Java for the smartphone application, Python for backend server and machine learning related tasks, and JavaScript for the smartwatch application and web application. This section will also briefly mention and describe a number of important frameworks utilized.

## **AngularJS**

AngularJS is one of the best web application frameworks developed and maintained by Google [26]. This framework has a number of useful built-in features for speeding up the development process. One of such important and useful features is data bindings. This allows a variable to be bound on a View element. This framework is used for developing the real-time web application.

## **Tornado**

Tornado is a Python web server which has built-in supports for handling HTTP requests as well as WebSocket connections [27]. This framework is used for developing the backend server for handling incoming sensory data from the smartphone and smartwatch for both activity recording for data collection and real-time monitoring use cases.

## **Scikit-learn, Pandas, and NumPy**

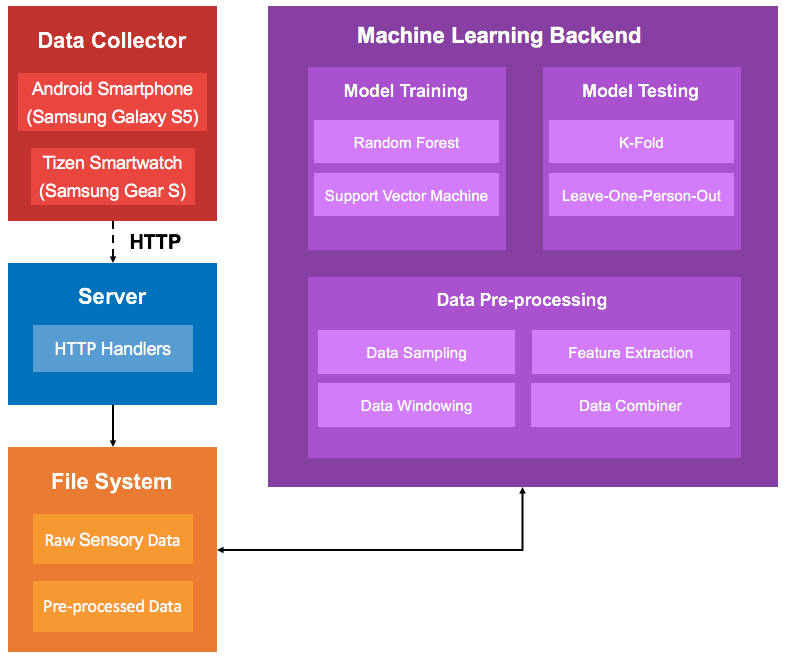
These three libraries are some of the well-known machine learning related libraries. Scikit-learn provides concrete implementations of various machine learning algorithms and some data manipulation tools such as K-Fold cross validation. Pandas provides a data structure which is very useful for manipulating data. NumPy provides implementations of many mathematical functions such as calculating correlations, Fast Fourier transform (FFT), etc.

## **MongoDB**

MongoDB is one the most famous NoSQL databases in the market right now. Unlike SQL database, MongoDB does not use any database schema. In this project, MongoDB is used for storing sensory data sent by the smartphone and smartwatch in the real-time activity monitoring use case.

# **Human Activity Recognition Methodology**

## **Overview of the Data Collection Components**



**Figure 1.** The data collection overall system architecture.

The data collection system consists of four main components. Those four components are:

1. **Data Collector**

There are two different devices used as sources of sensory data. The first device is an Android smartphone (Samsung Galaxy S5) and the second device is a Tizen smartwatch (Samsung Gear S). Both devices gather data from all accessible sensors.

1. **Server**

The server handles all communication between the smartphone and smartwatch as well as storing the collected sensory data sent by the two devices. Each raw sensory data received by the server is stored as CSV files in the file system.

1. **Machine Learning Backend**

This component is the most important component in this system as it handles all machine learning related logics. This includes raw sensory data pre-processing (data sampling, windowing, and feature extraction), model training, and model testing. There are different type of machine learning algorithms and testing methods involved and handled by this machine learning backend system.

1. **File System**

The file system is used by the server as a non-volatile file storage for the raw sensory data. Moreover, the machine learning backend system retrieves and pre-processes the raw sensory data. The resulting pre-processed data is stored back in the file system as well and will be used for model training and testing.

In this project, unlike the work done by the previous student in [9], features generated from barometer and gyroscope raw sensory readings will also be investigated as more complex activities are considered. In Section 6, there will be an analysis on whether the features generated from barometer and gyroscope data improve the performance of the model.

## **Data Collector**

There are two data collection devices involved in this project: a smartphone and a smartwatch. The smartphone is a Samsung Galaxy S5 running on Android 5.0 while the smartwatch is a Samsung Gear S running on a Tizen OS.

* + 1. **Android Smartphone Application**

The application is developed with the minimum SDK version of 16. The Android smartphone application is implemented to collect 6 different type of sensory data: accelerometer, linear accelerometer, gyroscope, barometer, magnetic, and gravity. Although every accessible sensory data is collected, not all sensors will be considered in the model training and testing.

In the implementation, the sensors are sampled for every 0.2 seconds which is the normal sampling rate provided and suggested by the Android sensor framework. Sampling the sensors every 0.2 seconds should be enough since having faster sampling rate may capture noises as well while having slower sampling rate may lose important readings. However, this sampling rate may not be followed strictly as the Android operating system may have other interruptions with higher importance resulting in a little bit more delay added to the sensor sampling. The additional delay will not affect the collected sensory data as there will be a data sampling pre-processing step in a later stage using a Python script.

Figure 2 shows the main user interface for the activity recording use case. This is used for controlling the sensory data collection from test subjects. Some important configurations can be set up from this user interface, for example, the server URL which the smartphone will be communicating with, the number of seconds after which the activity recording should stop, the unique ID which should identify the recently recorded sensory data, and the activity to be performed. Furthermore, sensory data can easily be sent to the backend server by just clicking the “Send Data” button.

<<FIGURE 2>>

In order to notify the test subject when the recording starts and stops, a sound signal is used. As soon as the “Start Recording” button is pressed, a single beep sound will be played, indicating that the activity recording has started and the test subject should start performing the activity immediately. When the activity recording is done, either because the “Stop Recording” button is pressed or the timer has ended, a triple beep sound will be played. Since the sensory data is recorded immediately after the “Start Recording” button is pressed, there will most likely be noisy data in the beginning of the recorded data. This will be discarded in the first data pre-processing step which will be discussed later.

* + 1. **Tizen Smartwatch Application**

The application for the smartwatch is developed for Tizen OS 2.2 using HTML. The SDK provides different APIs for collecting data from the sensors. There are 6 different sensory data collected: accelerometer, gyroscope, barometer, magnetic, light, and ultraviolet. Similar to the smartphone’s collected sensory data, not all sensory data collected from the smartwatch is considered in later stages. For example, it is very unlikely to distinguish different activities using data collected from the light and ultraviolet sensors.

The sensory data is also recorded for every 0.2 seconds. The sampling rate is chosen to be the same as the smartphone counterpart. However, unlike the Android sensor framework, Tizen OS does not have a sensor manager which handles the desired sampling rate. Instead, the JavaScript code has to keep track whether 0.2 seconds has passed since the last sensory data recording. Thus, the desired sampling rate may not be followed strictly due to interruptions like user interactions.

An important thing to note is that the smartwatch is not communicating with the smartphone via Bluetooth. The smartwatch actually supports Bluetooth connection with a Samsung smartphone. However, this feature is not well documented for the web-based application development. Thus, the smartphone and the smartwatch communicate with each other with the help of the server as an intermediary.

For each activity recording session, the smartphone only needs to communicate for 3 times with the smartwatch in general. The first time happens when the test subject presses on the “Start Recording” button on the Android application. The second time is when the test subject presses on the “Stop Recording” button on the Android application or when the timer runs out. The third time is when the “Send Data” button on the Android application is pressed. For each of these events, the Android application actually sends a HTTP request to the server and the server will notify the smartwatch using a WebSocket connection, allowing the smartphone to indirectly communicate with the smartwatch.

Figure 3 shows the user interface of the smartwatch application. The buttons are used only just in case the communication link between the smartphone and smartwatch is broken. There is also a textbox for specifying the server’s URL. There is a list of sensor readings which are updated in real time with the latest sensory readings.

<<FIGURE 3>>

* + 1. **Placement and Orientation of the Smartphone and Smartwatch**

The smartphone is put inside the right pocket, facing outward, and head-in first. This position is chosen so that the test subjects can easily press on the “Start Recording” button on the Android application placed near to the bottom of the display and push the smartphone directly into their pants. This position will reduce the amount of noisy data generated. However, there is no restriction imposed on the type of pants used resulting in various pocket sizes and shapes. Therefore, the smartphone may tilt during the activity recording session.

The smartwatch is placed on the test subject’s dominant hand. The dominant hand is chosen due to the fact that complex hand-based activities are considered in this project and people tend to use their dominant hand to perform these activities. Figure 4 better illustrates the position of the smartphone and the smartwatch.

<< FIGURE 4>>

* + 1. **Data Collection Steps**

## **Data Pre-processing**

Test

## **Machine Learning Algorithms**

Test

**Data Collection (5 pages max)**

* + - How the data collection is conducted (the steps done by the test subject, including the websocket communication part 🡪 use sequence diagram)

**Data Pre-processing (5 pages max)**

* + - A diagram to illustrate the pre-processing steps
    - Data sampling (explain why there is another data sampling 🡪 something to do with the inconsistent sampling frequency of the Android and Tizen apps)
    - Data windowing
    - Feature generation (explain that the features are generated for each window)
    - Data normalization (std or minmax?)

**Machine Learning Algorithm (3 pages max)**

* + - Mention the machine learning algorithm used in this project (which one is the main one and which one is just for comparison since the main one is better)
    - What is Random Forest and how it works?
    - What is SVM and how it works (cost, gamma, RBF kernels)?
    - Look for something as illustration (for RF especially, like how it works)

# **Results and Analysis**

# **Real Time Human Activity Recognition**

# **Issues and Challenges**

## **Human Activity Recognition**

As the human activity recognition technology relies mostly on machine learning algorithm, firstly, there will be a number of challenges related to data pre-processing steps. For example, a lot of experiments must be conducted to find the best data sampling rate and window size. Moreover, there will be a lot of time spent on understanding the machine learning classifier algorithms, especially the Support Vector Machine (SVM). Experiments will also be needed to find out the best hyper parameters for the SVM model training.

Another possible issue is related to the variability of how people use a smart-watch and their hand positions when performing certain activities. This may affect the sensor data readings and hence a more complex data pre-processing steps may be needed. Lastly, as there seems to be no previous research work on human activity recognition using both a smartphone and a smart-watch as complimentary smart devices, this may require efforts on synchronization. A new classification paradigm must be devised such that data from both devices can be used to improve the accuracy and to include more hand-based activities.

## **Real-time Recognition System**

The current human activity recognition model needs a stream of data segmented into data windows of equal length. Each data window must be long enough so that there is enough context to predict an activity, but short enough so that the activity can be recognized as soon as possible. Finding the balance between the prediction accuracy and the time needed to get a prediction is hard and needs to be determined by experiments. If the accuracy is high but the user needs to wait for too long, it is bad for the application’s user experience. On the other hand, if the user receives the prediction quickly but the prediction is not correct, the prediction result will be useless.

# **Approaches and Technologies**

## **Tools**

* + - 1. **Python**

Python is one of the most popular programming languages in the world nowadays. It is popular due to the fact that it is easy to learn, powerful, and supported by a huge community. Python will be used for training and predicting activities using the LibSVM library. As one of the project objectives is providing real time activity recognition, Python will also be used as a server and provide endpoints for the Android application.

* + - 1. **Android and Java**

An Android application will be needed since the accelerometer sensor data will be taken from an Android based smartphone. The Android framework includes APIs needed for collecting accelerometer sensor readings.

* + - 1. **Tizen-based Smart-watch**

As mentioned in the objectives, this project is going to incorporate sensor data taken from a smart-watch. This project will be using a Tizen-based smart-watch, a Samsung Gear S. A simple application for collecting sensor data will also be developed.

## **Data Pre-processing and Prediction Pipeline**

Data pre-processing plays a very important role in any machine learning projects, including this project. Currently, there is no data pipeline for data pre-processing and activity prediction. According to the previous student’s guidelines, each step is performed manually. As a result, it is very error prone and may slow down the development.

This project aims to build a data pipeline such that all data pre-processing steps including the activity recognition step is automated. Given a stream of raw sensor data, the pipeline will produce the activity recognized as the final results. Please note that this pipeline is only applicable for real-time human activity recognition system. That being said, however, the data sampling and feature extraction code can be used for pre-processing the training data as well. Figure 1 illustrates the data pipeline.

****

Figure 1: Data pipeline diagram.

# **Project Tasks**

## **Code Rewriting**

The code written by the previous student is currently very messy as it seems the previous student wrote the code for the purpose of research and model building. Since this project aims to create a simple application which is able to recognize human activities in real-time, rewriting the code from scratch is needed as it will increase the readability and help the next student who is going to extend the project in the future. The code rewriting will include the Android application as well as porting the Matlab codes into Python. Moreover, rewriting the existing code also helps me understand the existing algorithm and processes better along the way.

## **Building Data Pipeline**

The next important task is to build a data pipeline to streamline the data pre-processing which includes data sampling and feature extraction. As mentioned previously, this data pipeline is important as it can reduce the human error rate. I believe the previous student conducted all data pre-processing steps manually whenever new data are obtained, which is very tedious. By using a data pipeline, it will improve the development time as well.

## **Enhance the Android Application to Perform Real-time Recognition**

The Android application should be able to perform real-time activity recognition. This involves building a simple backend server which will read the serialized classifier model and use the model to predict an activity given a stream of sensor data. It is worth to mention the Android application will be rewritten as well since there are many unused lines of code. Thus, it is very hard to make changes to the existing code and will require a lot of time.

## **Data Collection and Model Building using Sensor Data from Smart-watch**

Once the basic foundation of the smartphone activity recognition is settled, the project will continue working on the smart-watch activity recognition process. This involves creating a simple Tizen application to read and send sensor data to the backend server. Next, there will be data pre-processing steps and model building as well as devising an algorithm to use sensor data from both smartphone and smart-watch to improve the prediction accuracy results.

## **Model Testing and Improvement**

This task focuses mainly on thorough testing using multiple data sets from various sources. Other improvements needed will also be conducted in this period. Since there is a lot of variables involved in this project, there are uncertainties involved and hence I could not mention other specific improvement tasks at the time this project plan is written.

# **Schedules**

****

Figure 2: Project schedule

# **References**

|  |  |
| --- | --- |
| [1] | National Population and Talent Division, “Population in Brief 2016”, 2016. [Online]. Available: <http://www.nptd.gov.sg/Portals/0/Homepage/Highlights/population-in-brief-2016.pdf>. [Accessed Jan. 6, 2017]. |
| [2] | American Association of Retired Persons, “Aging in place: A state survey of livability policies and practices”, 2011. [Online]. Available: <https://assets.aarp.org/rgcenter/ppi/liv-com/aging-in-place-2011-full.pdf>. [Accessed Jan. 6, 2017]. |
| [3] | A. H. Nasution and S. Emmanuel, “Intelligent Video Surveillance for Monitoring Elderly in Home Environments”, *2007 IEEE 9th Workshop on Multimedia Signal Processing*, Crete, 2007, pp. 203-206. |
| [4] |  |
| [3] | Wang, D., Tan, A. H., & Zhang, D. (2015, December). Non-intrusive robuset human activity recognition for diverse age groups. In 2015 *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)* (Vol. 2, pp. 368-375). IEEE. |
| [4] | Weiss, G. M., Timko, J. L., Gallagher, C. M., Yoneda, K., & Schreiber, A. J. (2016, February). Smartwatch-based activity recognition: A machine learning approach. In *2016 IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)* (pp. 426-429). IEEE. |
| [5] | Bao L., Intille, S. (2004). Activity Recognition from User-Annotated Acceleration Data. In *Proc. Pervasive* (pp. 1-17). |
| [6] | Sun, L., Zhang, D., Li, B., Guo, B., & Li, S. (2010, October). Activity recognition on an accelerometer embedded mobile phone with varying positions and orientations. In *International Conference on Ubiquitous Intelligence and Computing* (pp. 548-562). Springer Berlin Heidelberg. |
| [7] | Anguita, D., Ghio, A., Oneto, L., Parra, X., & Reyes-Ortiz, J. L. (2012, December). Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine. In *International Workshop on Ambient Assisted Living* (pp. 216-223). Springer Berlin Heidelberg. |
| [8] | Bhattacharya, S., & Lane, N. D. (2016, March). From smart to deep: Robust activity recognition on smartwatches using deep learning. In *2016 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops)* (pp. 1-6). IEEE. |

[1] "Population in Brief 2016,"Accessed on: 6 January 2017Available: <http://www.nptd.gov.sg/Portals/0/Homepage/Highlights/population-in-brief-2016.pdf>

[2] "Aging in Place: A State Survey of Livability Policies and Practices,"Accessed on: 6 January 2017Available: <https://assets.aarp.org/rgcenter/ppi/liv-com/aging-in-place-2011-full.pdf>

[3] S. Chernbumroong, S. Cang, and H. Yu, "A practical multi-sensor activity recognition system for home-based care," *Decision Support Systems,* vol. 66, pp. 61-70, 2014.

[4] Y. Nam and J. W. Park, "Child Activity Recognition Based on Cooperative Fusion Model of a Triaxial Accelerometer and a Barometric Pressure Sensor," *IEEE Journal of Biomedical and Health Informatics,* vol. 17, no. 2, pp. 420-426, 2013.

[5] A. H. Nasution and S. Emmanuel, "Intelligent Video Surveillance for Monitoring Elderly in Home Environments," in *2007 IEEE 9th Workshop on Multimedia Signal Processing*, 2007, pp. 203-206.

[6] H. Foroughi, B. S. Aski, and H. Pourreza, "Intelligent video surveillance for monitoring fall detection of elderly in home environments," in *2008 11th International Conference on Computer and Information Technology*, 2008, pp. 219-224.

[7] R. Rawassizadeh, B. A. Price, and M. Petre, "Wearables: has the age of smartwatches finally arrived?," *Commun. ACM,* vol. 58, no. 1, pp. 45-47, 2014.

[8] Q. R. Feng, "Human Activity Recognition and Tracking," 2014.

[9] W. J. Yi, "Human Activity Data Analytics," 2015.

[10] L. Breiman, "Random Forests," *Mach. Learn.,* vol. 45, no. 1, pp. 5-32, 2001.

[11] M. Shoaib, S. Bosch, O. D. Incel, H. Scholten, and P. J. Havinga, "Fusion of smartphone motion sensors for physical activity recognition," *Sensors (Basel),* vol. 14, no. 6, pp. 10146-76, Jun 10 2014.

[12] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine," presented at the Proceedings of the 4th international conference on Ambient Assisted Living and Home Care, Vitoria-Gasteiz, Spain, 2012.

[13] J. Lester, T. Choudhury, N. Kern, G. Borriello, and B. Hannaford, "A hybrid discriminative/generative approach for modeling human activities," presented at the Proceedings of the 19th international joint conference on Artificial intelligence, Edinburgh, Scotland, 2005.

[14] N. Ravi, N. Dandekar, P. Mysore, and M. L. Littman, "Activity recognition from accelerometer data," presented at the Proceedings of the 17th conference on Innovative applications of artificial intelligence - Volume 3, Pittsburgh, Pennsylvania, 2005.

[15] L. Sun, D. Zhang, B. Li, B. Guo, and S. Li, "Activity Recognition on an Accelerometer Embedded Mobile Phone with Varying Positions and Orientations," in *Ubiquitous Intelligence and Computing: 7th International Conference, UIC 2010, Xi’an, China, October 26-29, 2010. Proceedings*, Z. Yu, R. Liscano, G. Chen, D. Zhang, and X. Zhou, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 548-562.

[16] D. Wang, A. H. Tan, and D. Zhang, "Non-Intrusive Robust Human Activity Recognition for Diverse Age Groups," in *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, 2015, vol. 2, pp. 368-375.

[17] M. Shoaib, H. Scholten, and P. J. M. Havinga, "Towards Physical Activity Recognition Using Smartphone Sensors," in *2013 IEEE 10th International Conference on Ubiquitous Intelligence and Computing and 2013 IEEE 10th International Conference on Autonomic and Trusted Computing*, 2013, pp. 80-87.

[18] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, "Activity recognition using cell phone accelerometers," *SIGKDD Explor. Newsl.,* vol. 12, no. 2, pp. 74-82, 2011.

[19] S. Chernbumroong, A. S. Atkins, and H. Yu, "Activity classification using a single wrist-worn accelerometer," in *2011 5th International Conference on Software, Knowledge Information, Industrial Management and Applications (SKIMA) Proceedings*, 2011, pp. 1-6.

[20] B. Mortazavi *et al.*, "Can Smartwatches Replace Smartphones for Posture Tracking?," *Sensors,* vol. 15, no. 10, pp. 26783-26800, 2015.

[21] F. Ramos, A. Moreira, A. Costa, R. Rolim, H. Almeida, and A. Perkusich, "Combining Smartphone and Smartwatch Sensor Data in Activity Recognition Approaches: an Experimental Evaluation."

[22] B. S. Andreas Dengel, Sebastian Baumbach, Slim Abdennadher, "Human Activity Recognition: Using Sensor Data of Smartphones and Smartwatches," presented at the 8th Int'l Conference on Agents and Artificial Intelligence, Rome, Italy, 2016.

[23] G. M. Weiss, J. L. Timko, C. M. Gallagher, K. Yoneda, and A. J. Schreiber, "Smartwatch-based activity recognition: A machine learning approach," in *2016 IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*, 2016, pp. 426-429: IEEE.

[24] M. Shoaib, S. Bosch, O. D. Incel, H. Scholten, and P. J. Havinga, "Complex Human Activity Recognition Using Smartphone and Wrist-Worn Motion Sensors," *Sensors,* vol. 16, no. 4, p. 426, 2016.

[25] S. Bhattacharya and N. D. Lane, "From smart to deep: Robust activity recognition on smartwatches using deep learning," in *2016 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops)*, 2016, pp. 1-6: IEEE.

[26] Google. (11 January). *AngularJS - Superheroic JavaScript MVW Framework*. Available: <https://angularjs.org/>

[27] Tornado. (11 January). *Tornado Web Server - Tornado 4.4.2 documentation*. Available: <http://www.tornadoweb.org/en/stable/>