GOOD-WALK RECOGNITION USING ANDROID SMARTPHONE ACCELEROMETER WITH APPLICATION ON SENIOR PATIENTS

BY

WENBO YU

THESIS
Submitted in partial fulfillment of the requirements for the degree of Master of Science in Computer Science in the Graduate College of the University of Illinois at Urbana-Champaign, 2016

Urbana, Illinois

Adviser:
Professor Bruce R. Schatz
ABSTRACT

Good walk from one’s everyday activities can be used towards chronic disease diagnosis. Smartphones have become increasingly popular among people across ages. Properties including light weight, computationally powerful make smartphones ideal platforms for activity tracking and analysis. This work focuses on good walk recognition using smartphone accelerometer readings. The algorithms are validated with activity data collected from a large pool of healthy college students and senior patients. Softwares are implemented for walk recognition and pulmonary function evaluations, and are integrated to a pipeline as part of a sequence of activity data analysis.
To my parents, for their love and support.
ACKNOWLEDGMENTS

This project would not have been possible without the support of many people. Many thanks to my adviser, Prof. Bruce R. Schatz, who gave me insights into tackling the problem and provided me with valuable datasets for testing purposes. Also thanks to Qian Cheng who introduced me to this project, and shared with me his previous work to help me getting started, and Joshua Juen who guided me through the process of integrating the algorithms into a pipeline he developed. And finally, thanks to my parents, and numerous friends who endured this process with me, always offering support and love.
TABLE OF CONTENTS

CHAPTER 1 INTRODUCTION ........................................ 1
  1.1 Problem Description ........................................ 2
  1.2 Challenges .................................................. 2
  1.3 Thesis Organization ....................................... 3

CHAPTER 2 BACKGROUND .......................................... 4
  2.1 Datasets ...................................................... 4
  2.2 Tools ........................................................ 7
  2.3 Support Vector Machine .................................. 7
  2.4 Fast Fourier Transform .................................. 8
  2.5 Random Forest ............................................. 8

CHAPTER 3 RELATED WORK ........................................ 9

CHAPTER 4 ACTIVITY DATA ANALYSIS ......................... 11
  4.1 Patterns and Features of Different Activities .......... 11
  4.2 Walking Pattern Comparison Between Subjects of Differ-
          ent Age Groups and Physical Status ............... 16

CHAPTER 5 METHODS ............................................. 19
  5.1 Preprocessing ............................................... 19
  5.2 Classification Algorithms ................................ 19
  5.3 Features .................................................... 20

CHAPTER 6 EXPERIMENTS ......................................... 21
  6.1 Stationary vs. Moving ..................................... 21
  6.2 Walk vs. Stairs ............................................. 22
  6.3 Stairs, Walk, Stationary 3-class Learning Algorithm Clas-
          sification ............................................... 24
  6.4 Stairs, Walk, Stationary Classification With Standard De-
          viation Threshold and Learning Algorithm .......... 26

CHAPTER 7 IMPLEMENTATION .................................... 28
  7.1 Smartphone Application .................................. 28
  7.2 Backend Pipeline ......................................... 28
CHAPTER 1

INTRODUCTION

Smartphones are becoming increasingly popular and powerful. About 58% of American adults carry around computing devices with advanced mobile operating systems and loaded with sensors.[1] It is worth mentioning that 42% seniors of age between 50 to 64, and 16% of seniors of age 65 or older have access the Internet through mobile wireless devices.[2] In 2015Q2, among all mobile operating systems, Android dominated the market with share of 82.8%, following iOS with share of 13.9%.[3]

Latest smartphones are beyond communication tools. Sensors and components like accelerometers, GPS, gyroscope are becoming more and more common even among low-end phones.

Immersive mobile games, complex image manipulation software as well as all kinds of health promotion apps are now available on mobile operating systems. Portability and the variety of sensors make smartphones ideal platforms for building health related applications. There are 5,805 health promotion apps available on Apple Appstore by February 2010.[2] The idea of accessing health information using smartphones is becoming more and more widely accepted. According to a survey conducted in late 2012, 31% of mobile phone users access health information, which has increased significantly from 14% in 2010.[4]

Mobile apps make monitoring chronic diseases easier. Long term tracking is usually required to identify chronic diseases. With the rising popularity of smartphones people are carrying around powerful personal computing devices on daily basis, and thus tracking such diseases with smartphones introduces little extra cost to patients.
1.1 Problem Description

Good walk patterns can imply the health conditions of a person. For diseases like chronic obstructive pulmonary disease, 6-min walk test has been proven to be an effective diagnosis method [5].

However, to monitor pulmonary diseases, good walk from patients across a long period of time need to be collected and analyzed. It is a large time commitment for the patient to visit the medical institutes periodically to have their good walk recorded.

In this project, we aim to develop a good walk recognition mechanism which can filter on patients’ daily activities, and result in sessions of good walk they have taken during the day. We define good walk as long interval of natural walk. Accelerometer is the only sensor required. For the purpose of this study, we assume a person’s everyday activity fall into seven types (walking, jogging, upstairs, downstairs, sitting, standing, car riding). In chapter 4 we will look into each of the seven selected activity types, and compare walk patterns among different subject groups.

The daily activity information will be collected from the phone patients carry, and processed through a pipeline in the backend. There will be multiple tasks in this pipeline, in this project we focus on the first stage of the pipeline which is good walk recognition. Later stages in the pipeline include pulmonary function prediction, gait speed detection and oxygen saturation measurements.

Target users of this health monitoring system will be mostly senior patients. Considering their special physical conditions, we need to apply constraints and test on subjects that fit in our target demographics. In chapter 4, we analyze activity dataset with different demographics, and discuss how age and sickness affect one’s walk patterns.

1.2 Challenges

Majority of activity datasets are collected from young and healthy subjects, which are not the target users of monitoring system. Datasets from senior patients include limited number of subjects and little variety of activities.
1.3 Thesis Organization

In chapter 2, we will discuss the background information of the project including datasets and software used. In chapter 3, related work around activity recognitions will be discussed. In chapter 4, we present detailed analysis of the activity datasets. In chapter 5, we introduce methods used for activity classifications. In chapter 6, we will show experiments to prove effectiveness of our approach. In chapter 7, we will give a high level overview of the software implementation. Conclusions and future work will be presented in chapter 8 and 9.
CHAPTER 2
BACKGROUND

2.1 Datasets

2.1.1 Fordham Activity Prediction Dataset

Activity Prediction dataset was collected by Fordham University. It contains large amount of everyday activities collected from Android phones which subjects carry around in the lab[6].

Attributes of the dataset are listed below.

- Acceleration unit: 1 unit = 0.1\(g\) = 0.981\(m/s^2\)
- Activities: walking, jogging, upstairs, downstairs, sitting, standing
- Attributes: [user],[activity],[timestamp],[x-acceleration],[y-acceleration],[z-acceleration]
- Number of subjects: 36
- Positions: front pants leg pocket
- Platform: Android
- Sample data: 33,Jogging,49108272262000,-2.3018389,1.6889231,0.08172209;
- Sampling rate: 20Hz
- Subjects: college students

This dataset lacks detailed documentation of data collection procedures, demographics, and types of phones used. There are some publications potentially introduced data collection process of the Fordham Activity Prediction dataset vaguely.
• Cell Phone-Based Biometric Identification
  
  – Number of subjects: 36
  
  – Data collection procedure: A research oversees the data collection process. Subjects perform activities include walking, jogging, upstairs and downstairs with Android phone placed in their front pant pocket [7].
  
  – Note: the number of subjects matches that of the dataset we have in possession, so this paper is likely to be the source of the dataset. It does not mention phone type, age, detailed task descriptions.

• Activity Recognition using Cell Phone Accelerometers [6].
  
  – Number of subjects: 29
  
  – Data collection procedure: the same as above
  
  – Note: On the page (http://www.cis.fordham.edu/wisdm/dataset.php) where the Fordham Activity Prediction dataset was downloaded, it mentioned that this paper corresponds to the dataset. However this paper provides no extra information which we are looking for and the number of subjects does not match.

• Identifying User Traits by Mining Smart Phone Accelerometer Data
  
  – Number of subjects: 70
  
  – Data collection procedure: Subjects are asked to walk between 5 to 10 minutes. Android phones are placed in their pockets [8].
  
  – Age: Mostly college students aged between 18 and 24.
  
  – Device: $500 value phones provided by researchers
  
  – Note: this paper mentioned that it used data collected in Activity Recognition using Cell Phone Accelerometers, so it’s reasonable to assume that the subjects are also college students between the age of 18 and 24 in the fordham dataset.

• Limitations with Activity Recognition Methodology & Data Sets [9]
  
  – Number of subjects: 59
– Data collection procedure: subjects are asked to do a set of activities outdoor; phones are placed in subjects’ pants pocket [9].

– Note: it is very confusing that according to this paper the number of subjects in dataset collected in Activity Recognition using Cell Phone Accelerometers is 59, while that very paper itself claim to be 29.

2.1.2 Carle Dataset

In order to validate the effectiveness of recognition model on senior patients, we used the dataset which contains 28 senior patients and 10 regular people as subjects collected by Cheng et al. [10].

During the data collection process, subjects were asked to perform a 6-minute-walk back and forth on a 10-meter walkway. Afterwards, subjects may optionally choose to perform a passage of free-walk (without constraint on speed or time period) on a circle track. The activity data collection device (Motorola Droid Mini) was attached to subject’s lower back using fannypack [10].

More details about the dataset are listed below.

• Acceleration unit: 1 unit = 1 m/s²
• Activities: walking, stationary
• Attributes: [timestamp],[x-acceleration],[y-acceleration],[z-acceleration]
• Age: 50-80
• Device: Motorola Droid Mini
• Number of subjects: 28 senior patients, 10 regular people
• Positions: waist
• Platform: Android
• Sample raw data: 12:30:13.972, 9.02135181427002, 1.5131354331970215, 2.5091233253479004
• Sampling rate: 60 Hz
• Subjects: seniors patients, regular people

In this study we will train model using Fordham dataset and test on Carle dataset. Since the two datasets have different sampling rates (60Hz for Carle, 20Hz for Fordham), we preprocess the Carle dataset to convert its sampling rate to the same as that of Fordham, by including only every third sample in the processed Carle dataset.

2.2 Tools

2.2.1 scikit-learn

Scikit-learn is a Python programming language based machine learning library. It implements various classification algorithms including support vector machines, random forest and feature selection algorithms such as univariate feature selection, and recursive feature elimination [11]. In this study, we primarily used support vector machine and random forest modules.

2.2.2 Statsmodels

Statsmodels is a Python module which offers a wide variety of statistical analysis including linear regression model, discrete choice model and generalized linear model to facilitate exploring the data [12]. In this work we use Statsmodels to compute activity features.

2.3 Support Vector Machine

Support Vector Machine (SVM) is a scalable, efficient and theoretically sound learning algorithm. It finds the optimal hyperplane, which has the maximum distance between both classes to be classified by maximizing the following equation [13]

$$\min_{w \in \mathbb{R}^d, \xi_i \in \mathbb{R}^+} \frac{1}{2} w.w^T + C \sum_{i=1}^{N} \xi_i$$  \hspace{1cm} (2.1)
subject to
\[ y_i(w^T x_i + b) \leq 1 - \xi_i \text{ for } i = 1...N \] (2.2)

where \( y_i \) are classes, \( b \) is the offset and \( w \) is the norm to the hyperplane. Hinge loss is used as the loss function. Penalty parameter \( C \) is set to 1.0.

2.4 Fast Fourier Transform

Discrete Fourier transform (DFT) can convert a time series to frequency domain in \( 2N^2 \) computations. Fast Fourier transform (FFT) is a more efficient way to compute the DFT by reducing the computations needed to \( 2N \log N \) [14].

2.5 Random Forest

Decision Tree algorithms is know for suffering from the problem of overfitting [15]. Random Forest is a ensemble method developed to overcome this drawback of decision tree. It is a collection of decision trees which are generated by randomly selecting features at each node, voting scheme is used to decide the class of the input sample [15]. The number of trees in the forest is set to 10. Gini impurity is set as criterion. The maximum number of features to be evaluated when performing a split is \( \sqrt{\text{number of features}} \).
Activity recognition has become an increasingly popular topic. Many studies focus on activity recognition software built for mobile applications [16, 6, 17].

There are three types of activity recognition models, impersonal (train and test on different subjects), personal (train and test on the same subjects), hybrid (training subjects and testing subjects have overlaps) [9]. For our use case, the impersonal model is the most appropriate because training data provided by users are likely to be unreliable. The Fordham Actitracker dataset contains activity data labeled by users, which turns out are mostly falsely labeled.

As of classification algorithms, SVM is one of the most popular [18, 19, 20]. Ravi et al. studied activity recognition using accelerometer data collected [20]. The features selected are mean, standard deviation, energy (calculated from FFT magnitudes) and correlation. Activities collected include standing, walking, running, upstairs, downstairs, sit-up, vacuuming, brushing teeth. There are four different experiment setups, including impersonal setup which train and test on different subjects. Among eighteen learning algorithms applied, SVM was the one with the best accuracy of 73.33%.

Cheng et al. [21] in their work presented a novel activity data preprocessing approach. Accelerometer readings collected are first converted into earth fixed coordinate system using gravity vector collected from the sensor. A 10 Hz low pass filter is then applied to remove noise and improve the quality of signal collected with phone loosely attached to the body.

Kwapisz et al. [6] collected and labeled daily activities including walking, jogging, stairs, standing, sitting from 29 subjects. They discussed features of each activities and compared performances of using J48, Logistic Regression, Multilayer Perceptron and Straw Man with 10-fold validation. The features they selected are average, standard deviation, average absolute difference, average resultant acceleration, time between peaks and binned distribution.
Brajdic & Harle [22] evaluated various walk detection methods on a large dataset collected from 27 subjects, in six different positions (hand, hand with interactions, backpack, handbag, pants back pocket and pants front pocket), walk in different speeds. Both time domain and frequency domain algorithms are evaluated. Hidden Markov and K-Means are used from feature clustering. Results show that thresholding on standard deviation (threshold set to as 0.6 on window size of 0.8 s) and acceleration energy (threshold set to as 0.04 on window size of 1 s) are the two best algorithms. The walk detection is only effective when the activities to be recognized contain only walk and stationary activities (pickup phone, typing, rotate on office chair, transition from standing and walking and idle). Our project requires a recognizer that can identify walk from everyday activities including other moving activities, e.g., upstairs and downstairs, with high precision. In later chapters we will analyze different activities and show why standard deviation threshold of 0.6 can not recognize walking from other moving activities.
CHAPTER 4

ACTIVITY DATA ANALYSIS

In this chapter we analyze and display attributes of different activities, as well as the same activity performed by different types of subjects.

4.1 Patterns and Features of Different Activities

In this section we analyze the seven different activities (walking, jogging, upstairs, downstairs, sitting, standing, car riding) by observing their patterns and features calculated. The following data are from the Activity Prediction Dataset, which has userid, activity, timestamp, x-acceleration, y-acceleration and z-acceleration as attributes.

For the purpose of good walk recognition, we want to isolate walk pattern from the various orientations the phone might be in. Since the Activity Prediction Dataset does not have gravity distribution available as an attribute, it is impossible to eliminate the effect of orientation from accelerations on a single axis. Thus we will be looking into the magnitude of acceleration vector instead.

\[
magnitude = \sqrt{x_{\text{accel}}^2 + y_{\text{accel}}^2 + z_{\text{accel}}^2}
\]

The features we used are inspired by the work of Cheng et al. in their previous study of oxygen saturation prediction and health monitoring [10, 21]. Gait features selected are mean, standard deviation, mean crossing rate, root mean square, autocorrelation coefficient, coefficient of variance, peak frequency, Shannon entropy, subban energy and subban energy ration. The machine learning model built using those features can effectively predict oxygen saturation.

We computed 10 features for the seven aforementioned activities and shown in 4.1. The 10 selected features are
• Stand Deviation (STD): a measure of dispersion in a dataset, it implies
the intensity of activity

• Root Mean Square (RMS): square root of the mean of the squares of
the values

• Autocorrelation Coefficient: a measure of correlation between the past
and future of a time domain dataset, it implies how repeatable the
dataset is

• Mean Crossing Rate (MCR): the rate of the signal changes from below
mean to above mean

• Mean: the average of magnitude of acceleration vectors

• Peak Frequency (PF): the dominant frequency determined by the mag-
nitude of FFT result

• Shannon Entropy: diversity measurement of the dataset

• FFT coefficients: frequency domain representation of the dataset

• Cadence: step per minute, calculated by using step counting algorithm
developed by Cheng et al. [21].

• Coefficient of variance (CV): dispersion measurement of frequency dis-
tribution

As can be observed from both the figure 4.1 and Peak Frequency in the
table, jogging has the highest frequency among all non-stationary activities
and the highest magnitude among all activities. Its patterns and features
look similar to that of walking in table 4.2.

Upstairs and downstairs in comparison have smaller mean, standard devi-
ation and root mean square. Their patterns also look different from walking
and jogging. It can be observed that the peak frequency of downstairs is
higher than both walking and upstairs as it is the most effortless activity
among the three. Upstairs and downstairs both have average standard devi-
ation of over 3.5. If 0.6 is used as threshold, a method introduced by Brajdic
& Harle [22], to recognize walk from a dataset which also includes stairs, it
is very likely that stairs activities will be mis-classified as walk.
Standing and sitting are the two stationary activities. From figure 4.6 and figure 4.5 we can observe that the signals are considerably more flat compared to that of other activities, which explains the distinctively low coefficient of variance, and standard deviations of 0.165 and 0.085 versus larger than 3.800 for all other activities. The means of both activities are around 9.6 m/s\(^2\), with the earth’s gravity set to 9.81 m/s\(^2\).

Car riding is a very common activity for seniors. Relative to the vehicle, riding is a stationary activity and is the same as sitting in terms of movement of the person carrying the phone. The recorded acceleration readings are relative to the Earth, and to which the person in the vehicle is moving at high speed and varying accelerations.

We collected a small dataset of a 20-minute car ride. The phone was placed in the subject’s front pant pocket. The car went across different driving conditions such as school area and freeway. Several stop signs and red lights were encountered.

Figure 4.7 shows a passage of 3-minute car ride in the dataset. The signal is distinctively different from those of other activity, as it displays no periodicity, changes abruptly. The acceleration magnitude also have a smaller range than other moving activities (walking, jogging, upstairs and downstairs), because a car ride is more steady overall.
Figure 4.2: Walking

Figure 4.3: Upstairs

Figure 4.4: Downstairs
Figure 4.5: Sitting

Figure 4.6: Standing

Figure 4.7: Car Riding
Table 4.1: Features Calculated for Seven Activities

<table>
<thead>
<tr>
<th></th>
<th>Walking</th>
<th>Jogging</th>
<th>Upstairs</th>
<th>Downstairs</th>
<th>Standing</th>
<th>Sitting</th>
<th>Riding</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cadence</td>
<td>0.515</td>
<td>0.823</td>
<td>0.753</td>
<td>1.01</td>
<td>0.671</td>
<td>0.497</td>
<td>0.527</td>
</tr>
<tr>
<td>STD</td>
<td>4.500</td>
<td>6.801</td>
<td>3.871</td>
<td>4.197</td>
<td>0.165</td>
<td>0.086</td>
<td>0.324</td>
</tr>
<tr>
<td>MCR</td>
<td>62.429</td>
<td>107.976</td>
<td>89.818</td>
<td>100.523</td>
<td>135.742</td>
<td>128.160</td>
<td>71.044</td>
</tr>
<tr>
<td>RMS</td>
<td>1.233</td>
<td>1.518</td>
<td>1.145</td>
<td>1.170</td>
<td>0.981</td>
<td>0.986</td>
<td>1.019</td>
</tr>
<tr>
<td>AC</td>
<td>0.649</td>
<td>0.566</td>
<td>0.373</td>
<td>0.332</td>
<td>0.210</td>
<td>0.217</td>
<td>0.297</td>
</tr>
<tr>
<td>CV</td>
<td>0.401</td>
<td>0.527</td>
<td>0.367</td>
<td>0.393</td>
<td>0.017</td>
<td>0.009</td>
<td>0.0342</td>
</tr>
<tr>
<td>PF</td>
<td>1.979</td>
<td>2.626</td>
<td>1.708</td>
<td>2.186</td>
<td>4.422</td>
<td>3.191</td>
<td>3.220</td>
</tr>
</tbody>
</table>

4.2 Walking Pattern Comparison Between Subjects of Different Age Groups and Physical Status

In this section we take a closer look at the walk patterns of subjects in Fordham dataset and Carle dataset. We show how disease and age affects one’s walking patterns.

4.2.1 Walking Comparison Between Healthy Senior Subjects and Disease Affected Senior Subjects

Figure 4.8 and 4.9 show walking pattern of a disease affected Senior subject and a healthy senior subject respectively. The signal from patient has larger distance between peak (0.7s vs. 0.5s), smaller range of acceleration ($15-7m/s^2$ vs. $16-5m/s^2$). The signal also appears to be less smooth.

In table 4.2, we computed activity features of different subject groups. Compared to healthy senior subjects, disease affected senior subjects have lower cadence and PF, which implies less steps taken per minute; lower mean, standard deviation, CV and RMS, which implies the movement is less significant; lower AC, higher Shannon Entropy and higher MCR, which implies that the movement is less steady.
4.2.2 Walking Comparison Between Healthy Senior Subjects and Young Regular Subjects

We have seen the differences of walking pattern between regular subjects and patient subjects of the same age group. In this section we explore how age affect one’s walking. Figure 4.9 shows walking pattern of a healthy senior subject between the age of 50-60, and 4.2 shows the pattern from a healthy college student between the age of 18-24. Both signals have similar distances between peaks, while college student’s signal has larger range of acceleration.

Compared to healthy young subjects, healthy senior subjects have lower cadence and PF, which implies less steps taken per minute; lower mean, standard deviation, CV and RMS, which implies the movement is less significant; higher Shannon Entropy, MCR, which implies that the movement is less steady. It is notable that the AC of healthy senior is higher, despite having higher entropy and MCR. Higher AC means that the signal has stronger periodicity, and usually comes with lower MCR and entropy.
Figure 4.9: Healthy Senior Subject Walk

Table 4.2: Walk Attributes of Different Subject Groups

<table>
<thead>
<tr>
<th></th>
<th>Disease Affected Senior</th>
<th>Healthy Senior</th>
<th>Healthy Young</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cadence</td>
<td>0.449</td>
<td>0.502</td>
<td>0.515</td>
</tr>
<tr>
<td>Mean</td>
<td>9.631</td>
<td>9.670</td>
<td>11.22</td>
</tr>
<tr>
<td>STD</td>
<td>1.142</td>
<td>2.298</td>
<td>4.500</td>
</tr>
<tr>
<td>MCR</td>
<td>49.309</td>
<td>44.552</td>
<td>62.429</td>
</tr>
<tr>
<td>RMS</td>
<td>0.989</td>
<td>1.014</td>
<td>1.233</td>
</tr>
<tr>
<td>AC</td>
<td>0.711</td>
<td>0.807</td>
<td>0.649</td>
</tr>
<tr>
<td>CV</td>
<td>0.118</td>
<td>0.237</td>
<td>0.401</td>
</tr>
<tr>
<td>PF</td>
<td>1.570</td>
<td>1.886</td>
<td>1.979</td>
</tr>
<tr>
<td>Entropy</td>
<td>5.945</td>
<td>5.710</td>
<td>6.100</td>
</tr>
</tbody>
</table>
In this chapter we introduce the methods we used to recognize good walk sessions.

Our target users are pulmonary patents who are capable of doing limited types of activities due to their physical conditions. We made the following assumptions: 1) users are capable of sitting, standing, walking, upstairs and downstairs. 2) the phone used for collecting accelerometer readings is positioned close and tight to the user’s body.

Based on the observations made in Chapter 4, we created three classes for our classification task. Walk class includes walking and jogging, stairs class include upstairs and downstairs, stationary class includes sitting and standing.

The users are expected to carry the phone most of the time, and the aggregated activity data collected for filtering will be large. Thus the goal of the recognizer is to recognize good walk with high precision and reasonable recall.

5.1 Preprocessing

Each sample contains accelerations on three axes, which are used to calculated the magnitude. A session is the smallest unit for activity recognition, it is configured to contain acceleration magnitudes of 10-second activity. The sliding is set to 50%, so every session overlaps the next by 5 seconds.

5.2 Classification Algorithms

The machine algorithms we used in the project are SVM and Random Forest. The models are implemented in scikit-learn and default parameters
were used. For SVM we used LinearSVC class, which is implemented with liblinear and is capable of handling large samples efficiently. We used the RandomForestClassifier class in scikit-learn as Random Forest model.

Thresholding on features such as standard deviation is a potentially effective activity recognition algorithm [22]. We use this approach to classify stationary activities from moving activities.

5.3 Features

The features we selected are cadence, standard deviation, root mean square, autocorrelation coefficient, mean crossing rate, mean, peak frequency, Shannon entropy, FFT coefficients and coefficient of variance. In chapter 4, we discussed how each feature can be used to characterize activity sessions.
CHAPTER 6

EXPERIMENTS

In this section we use the leave-subject-out cross-validation scheme to evaluate our methods on Fordham Activity Prediction dataset. We also build models using Fordham Activity Prediction dataset as training sets and evaluate the models by testing on Carle dataset.

6.1 Stationary vs. Moving

The standard deviation of acceleration magnitudes in a 10-second session is used as the feature for distinguishing stationary from moving. Standard deviation thresholds from 0.1 to 1.9 are tested on datasets collected using college students, healthy seniors and disease affected seniors as subject groups.

In Fordham Activity Prediction dataset (which subjects are healthy college students), walking, jogging, upstairs and downstairs are labeled as moving class. Sitting and standing are labeled as stationary class. Thresholds between 0.4 and 1.9 achieved perfect precision and recall, as shown in figure 6.3.

In Carle dataset (which subjects contain healthy and disease affected seniors), 6-minute-walk and free-walk are labeled as moving and the rest of the samples are labeled as stationary. For healthy seniors, perfect moving recognition precision and recall are achieved with standard deviation threshold set to 1.6, as shown in figure 6.2. For disease affected seniors, perfect moving recognition precision is achieved with standard deviation threshold set to 1.4, as shown in figure 6.1.

In table 4.2 we computed activity features of different subject groups. It shows that the average standard deviation of seniors walking is smaller than that of college students. Figure 6.4 compares moving recognition precisions with threshold set to different values on datasets collected from different
subject groups. It can be observed that in order to achieve the same precision, threshold on seniors need to be higher than on college students, which is counter intuitive given the fact that seniors have lower walking standard deviation. One possible explanation is that the Carle dataset was not eliminated from the noisy sensor readings collected during the setup and finishing stages which are labeled as stationary. The noisy signals usually have high magnitudes and standard deviation, which make them likely to be misclassified into moving class.

The average standard deviation of disease affected seniors is 1.142. If a standard deviation threshold above 1.14 is set for moving vs. stationary classification, many of walking sessions from disease affected seniors will not be captured. Thus a proper standard deviation threshold should be picked in between 0.7 (100% precision for college students, 96% for disease affected seniors, 96% for healthy seniors) and 1.1 (100% precision for college students, 97% for disease affected seniors, 98% for healthy seniors).

Higher standard deviation threshold boosts precision at the cost of lower recall, as shown in Figure 6.1, 6.2 and 6.3. For application on senior patients, lower STD threshold (0.7) should be picked. Compared to threshold set to 1.1, the precision drops by 1% while recall increases by 28%.

We applied standard deviation threshold on car ridings, results show that over 91% of the activity can be recognized as stationary with standard deviation threshold set to as larger than or equal to 0.7.

6.2 Walk vs. Stairs

After an activity session is classified into moving class, we need to further check whether it is walk or stairs.

The cross-validation results on Fordham Activity Prediction dataset are shown in table 6.1. Random forest outperforms SVM in terms of both walking prediction precision (96% vs. 94%) and stairs prediction precision (86% vs. 78%), while SVM makes prediction faster than random forest (0.082s vs. 0.017s).

Since Carle dataset does not record subjects climbing up and down stairs, we are only able to test whether the subjects’ walking can be classified as walk class. Random forest recognized 19% more walking sessions than SVM.
Figure 6.1: Precision and Recall of Moving Recognition with Different STD as Thresholds on Disease Affected Seniors

Figure 6.2: Precision and Recall of Moving Recognition with Different STD as Thresholds on Healthy Seniors
Figure 6.3: Precision and Recall of Moving Recognition with Different STD as Thresholds on Healthy Young

Table 6.1: Walk vs. Stairs Prediction Cross Validation Results on Fordham Dataset

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>SVM</th>
<th>Random Forest</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Walk</td>
<td>Stairs</td>
</tr>
<tr>
<td>Precision</td>
<td>0.94</td>
<td>0.78</td>
</tr>
<tr>
<td>Recall</td>
<td>0.89</td>
<td>0.86</td>
</tr>
<tr>
<td>Support</td>
<td>4120</td>
<td>1770</td>
</tr>
</tbody>
</table>

(63% vs. 44%), as shown in table 6.2.

6.3 Stairs, Walk, Stationary 3-class Learning Algorithm Classification

In this approach, instead of using standard deviation as threshold to filter out stationary activities, we include stationary as the third class in the learning model.

The precisions of walk recognition using the 3-class model (0.92 with SVM,
Figure 6.4: Precision and Recall of Moving Recognition with Different STD as Thresholds on Different Subject Groups

Table 6.2: Walk vs. Stairs Prediction Results on Carle Dataset with Model Trained with Fordham Dataset

<table>
<thead>
<tr>
<th>Model</th>
<th>Walk</th>
<th>Stairs</th>
<th>Walk</th>
<th>Stairs</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Precision</td>
<td>0.44</td>
<td>NA</td>
<td>0.63</td>
<td>NA</td>
</tr>
<tr>
<td>Recall</td>
<td>2104</td>
<td>0</td>
<td>2104</td>
<td>0</td>
</tr>
</tbody>
</table>
Table 6.3: Stationary vs. Walk vs. Stairs 3-Class Learning Model Prediction Cross Validation Results on Fordham Dataset

<table>
<thead>
<tr>
<th></th>
<th>SVM</th>
<th>Random Forest</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Walk</td>
<td>Stairs</td>
</tr>
<tr>
<td>Precision</td>
<td>0.92</td>
<td>0.78</td>
</tr>
<tr>
<td>Recall</td>
<td>0.90</td>
<td>0.98</td>
</tr>
<tr>
<td>Support</td>
<td>4120</td>
<td>1700</td>
</tr>
</tbody>
</table>

Table 6.4: Stationary vs. Walk vs. Stairs Prediction Results on Disease Affected Seniors with 3-class Learning Model Trained with Fordham Dataset

<table>
<thead>
<tr>
<th></th>
<th>SVM</th>
<th>Random Forest</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Walk</td>
<td>Stairs</td>
</tr>
<tr>
<td>Precision</td>
<td>1.00</td>
<td>NA</td>
</tr>
<tr>
<td>Recall</td>
<td>0.05</td>
<td>NA</td>
</tr>
<tr>
<td>Support</td>
<td>2104</td>
<td>NA</td>
</tr>
</tbody>
</table>

0.96 with Random forest as shown in table 6.3) is similar to using the 2-class model (0.94 with SVM, 0.96 with Random Forest) on Fordham Activity Prediction dataset. Random forest achieves higher precision also on stairs and stationary recognition.

Model built with SVM achieves higher walk class prediction precision (100% vs. 98%) at the cost of significantly lower recall (5% vs 56%) as shown in table 6.2. Although high walk class prediction recall is not one of the targets, we still expect the value to be within reasonable range, and 5% is clearly unacceptable.

6.4 Stairs, Walk, Stationary Classification With Standard Deviation Threshold and Learning Algorithm

In this approach, we combined the standard deviation threshold with learning algorithm. First, we filter the activity session by setting 0.7 as standard deviation threshold. If standard deviation of the activity session is less than 0.7, it is classified into stationary class. If not, we pass the activity session as input into stairs vs. walk random forest/SVM based learning model, the
Table 6.5: Stationary vs. Walk vs. Stairs Prediction Results on Disease Affected Seniors with Model Trained using Fordham Dataset and Standard Deviation Threshold

<table>
<thead>
<tr>
<th></th>
<th>SVM</th>
<th>Random Forest</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Walk</td>
<td>Stairs</td>
</tr>
<tr>
<td>Precision</td>
<td>1.00</td>
<td>NA</td>
</tr>
<tr>
<td>Recall</td>
<td>0.11</td>
<td>NA</td>
</tr>
<tr>
<td>Support</td>
<td>2104</td>
<td>NA</td>
</tr>
</tbody>
</table>

output of the classifier is the final classification result.

Table 6.5 shows the result of applying the algorithm on senior patients. Compared to 3-class learning model classification (table 6.4), we observe improvements in walk class recognition precisions. The improvements are especially significant when random forest is used.
CHAPTER 7
IMPLEMENTATION

7.1 Smartphone Application

We use MoveSense, an Android phone application developed by Juen et al.[23], to collect activity data from users. The phone app passively monitors user’s daily activities, and transmits data to the backend pipeline for advanced analysis. Battery and mobile data is a major bottle neck of the phone app. We expect our users to have the application running 24 hours a day on a daily basis, significant battery life decrement can discourage the use from keeping the app installed and turned on. To avoid over draining battery and mobile data, an energy efficient algorithm is implemented to detect whether the phone is moving, and the app only transmits data when movement is detected.

7.2 Backend Pipeline

After movement is detected, the backend receives data transmitted from the phone app and stores it on the server. A pipeline process is started to analyze the data and finally decide diagnosis of disease.

We wrapped our algorithms in Python classes and made them ready to be integrated in the backend pipeline.

The classes we implemented are

- WalkAnalytics.py: implements walk vs. non-walk recognition, using standard deviation threshold method and stairs vs. walk random forest model combined
- WalkStairsAnalytics.py: implements random forest based walk vs. stairs model
• MoveStationaryAnalytics.py: implements standard deviation threshold

• PulmonaryAnalytics.py: implements pulmonary function evaluation model

The interface requires the following methods be implemented

• train_model()
  This function reads in the training dataset stored on the server, pre-processes the dataset by cutting it into sessions with overlapping and extracts features and labels. The features and labels are provided as inputs to the model generation function. Finally the model is exported to the server using joblib which is part of the scikit-learn library.

• load_model()
  This function imports the model from the server and set it as an instance variable.

• conduct_prediction(input_file, sensors, collection_frequency)
  This function preprocesses the input file and returns a list of prediction results.

• get_type()
  This function specifies the data type of the classification class.
CHAPTER 8
CONCLUSIONS

In this work we analyzed and compared between Fordham Activity Prediction dataset and Carle dataset to learn signals generated form different activities as well as from different subject groups. Based on observations, we grouped common daily activities into three classes. We showed the difference between stationary activities (standing, sitting, car riding) and moving activities (walking, jogging, upstairs, downstairs) in terms of various features computed. We showed differences and similarities among walking patterns from healthy young, healthy senior and disease affected senior.

We introduced models built for recognizing activities and evaluated their performances using datasets collected from various subject groups.

For stationary vs. moving recognition, we find using standard deviation between 0.7 (100% precision for college students, 96% for disease affected seniors, 96% for healthy seniors) and 1.1 (100% precision for college students, 97% for disease affected seniors, 98% for healthy seniors) as threshold achieves the best overall moving class recognition precision.

For stairs vs. walk recognition, our model built based on random forest achieved 96% walk class recognition precision and 86% stairs class recognition precision. It was also able to recognize 63% of walk from disease affected seniors.

For walk vs. non-walk recognition, combining standard deviation threshold method and stairs vs. walk random forest model outperforms stairs, walk and stationary 3-class learning model classification. It predicts Carle dataset as walk class with 98% precision and 72% recall.

We produced software in Python with interface required by the pipeline. The software includes walk recognition task and pulmonary function evaluation task.
CHAPTER 9

FUTURE WORK

9.1 Position Recognition and Conversion

Pulmonary function prediction works best with good walk information recorded from device attached to the waist. Unfortunately both Fordham Activity Prediction dataset and Carle dataset are collected with devices attached to the same position. Due to limitation of time frame we could not recruit people and collect activities in variety of positions. For future work, we could collect walk, stairs and stationary activities performed in different position such as pants pocket, coat pocket, waist, hands, backpack.

To ensure the walk recognized are all collected from the waist, we could 1) filter out all walk collected from undesired locations 2) convert signal to waist from other locations. Approach 1 would be straight forward and could be developed by reusing most of our current setup. Approach 2 would be more challenging both algorithm wise and data collection wise. The subjects would need to carry multiple devices attached to different locations at the same time and perform activities. Devices need to be identical and the data collection process need to be synchronous across all devices. Carrying multiple devices might interfere with normal activity movements, and could even be burdensome for certain subjects.

We recruited 6 healthy college students to study their walk data with phone attached to different positions, including waist, pants pockets, coat pockets, backpacks and hands. Subjects are asked to walk around a circle track with 10-meter diameter at their comfortable pace for 2-3 minutes. The following are attributes of the data collected.

- Acceleration unit: 1 unit = 1\text{m/s}^2
- Activities: walking, jogging, cycling, stationary, going up the stairs
• Attributes: \([x\text{-acceleration}] \ [y\text{-acceleration}] \ [z\text{-acceleration}] \ [\text{time from previous sample}]\)

• Number of subjects: 6

• Positions: waist, pants pockets, coat pockets, backpacks, hands

• Platform: Android

• Sample raw data: 0.902 5.276 10.439 20

• Sampling rate: 50Hz

• Subjects: college students aged 21-25

• Device: Galaxy S7 Edge

Using FFT coefficients as features, we are able to achieve 92% precision and 90% recall on classifying walk sessions into waist class using a SVM based model.

9.2 Activity Recognition With Gravity Sensor

Knowing how subject moves in the earth fixed coordinate system could be very helpful to recognize certain activities. For walk activity, we expect larger standard deviation of accelerations on the horizontal axis. And for stairs, accelerations on vertical direction should have the larger standard deviation as the movement is mostly vertical.

In Fordham Activity Prediction dataset, each sample includes accelerations on xyz axes which capture horizontal, vertical and forward-backward movement. However, the three axes are relative to the subject’s leg, which means a model trained with features extracted from three individual axes works only if the user orientate the phone in the identical way as in the training dataset.

For future work, we could collect gravity sensor readings to know the distribution of gravity on axes relative to the phone, and use which to convert the accelerometer readings to earth fixed coordinate system. Cheng et al. [21] in their work described in detail about how to convert accelerations onto earth fixed coordinate system using gravity vector.
9.3 Good-walk Recognition on Daily Activities Collected

The training datasets and testing datasets we used in this study are all generated in lab settings. Considering eventually the algorithms will be applied to recognize walk from full-natural activities, we need to further evaluate the models on datasets collected from subjects moving around doing their daily activities unsupervised and unrestricted.

Figure 9.1 shows a passage of daily activity from one of the developers collected using an Android app called Accelerometer Analyzer with phone positioned around waist. Green line indicates acceleration magnitudes. Blue line is the step plot of classification results, where value of 0 indicates non-walk class, and value of 50 indicates walk class. Fifty is an arbitrary number chosen in order to conveniently overlay the prediction results on acceleration. Text boxes on the top of the figure shows activities the subject was doing and their durations. The classification results look promising as the model does not mis-classify some common daily activities, such as cleaning the room and cooking, as good walk.

Figure 9.2 shows a passage of daily activity from one of the developers collected using the passive monitor Android app we developed with the phone positioned in shorts pocket. Green line indicates acceleration magnitudes. Blue line is the step plot of classification results, where value of 0 indicates stationary class, value of 30 indicates stairs class and value of 50 indicates walk class. The subject performed activities such as driving, standing, sitting, walking and stairs for over two hours. The passive monitor filters out stationary activities by applying standard deviation threshold, and push moving activities to the model to further check whether activity is walk.

A detailed documentation of durations and types of activities performed in Figure 9.2 is not available. Out of 198 10-second sessions, 34 of which are classified as stationary, 110 as stairs and 54 as walk. The model classifies the majority of activities as stairs class, however up/down stairs only account for a small portion of actual activities performed. The reason may be that the walks performed are not considered as good enough by the model. As shown in the previous example (Figure 9.1), activities such as cooking and cleaning, which are partially walking around, are not recognized as walk class. Since those activities clearly belong to moving class and do not qualify as good
walk, they are recognized as stairs. Low recall on stairs recognition is not a major concern, because for our application we only need to ensure that the model can recognize reasonable amount of good walk with high precision.

The good-walk recognition model used in both applications is standard deviation threshold method and stairs vs. walk random forest combined (Algorithm 1). Given features extracted from an activity session, we first compare the standard deviation with selected threshold (0.7), if it is less than the threshold, the session is classified as stationary. Otherwise we query the stairs vs. walk random forest model for classification result.

**Algorithm 1** Standard Deviation Threshold and Learning Algorithm Mixed Model for Good Walk Recognition

```
procedure WalkClassification
    data ← activity session features
    std ← activity session acceleration magnitude standard deviation
    threshold ← 0.7
    clf ← Walk vs. Stairs random forest classifier
    if std < threshold
        return Stationary
    else
        return clf.predict(data)
    end if
end procedure
```
Figure 9.1: Good Walk Recognition on Daily Activities

Figure 9.2: Good Walk Recognition on Daily Activities Collected from Passive Monitor
REFERENCES


