USER, DEVICE, ORIENTATION AND POSITION INDEPENDENT HUMAN ACTIVITY RECOGNITION ON SMART PHONES

by

Yunus Emre Üstev

B.S., Computer Engineering, İstanbul Technical University, 2008

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Computer Engineering

Boğaziçi University

2015

USER, DEVICE, ORIENTATION AND POSITION INDEPENDENT HUMAN ACTIVITY RECOGNITION ON SMART PHONES

APPROVED BY:

Prof. Cem Ersoy .....
(Thesis Supervisor)

Assist. Prof. Özlem Durmaz İncel.....
(Thesis Co-supervisor)

Assist. Prof. Bert Arnrich .....

Assist. Prof. Arzucan Özgür .....

Assist. Prof. Albert Ali Salah .....

DATE OF APPROVAL: 19.02.2015

ACKNOWLEDGEMENTS

First of all, I would like to express my deepest gratitude to my thesis coadviser Assist. Prof. Ozlem Durmaz Incel. I am appreciative for her continuous support, efforts and patience since the beginning of this thesis. I have been amazingly fortunate to meet her and it was a great opportunity for me to work with her during this process. I am also grateful to my thesis supervisor Prof. Cem Ersoy not only for his guidance during this thesis, but also for his continuous support since the beginning of my graduate study. It has been a great honour for me to run my graduate thesis with him.

Besides my advisors, I would like to thank the rest of my thesis jury: Assist. Prof. Bert Arnich, Assist. Prof. Arzucan Özgür, and Assist. Prof. Albert Ali Salah, for their insightful comments.

I would like to thank my friends and colleagues at NETLAB, especially Can, Halil and Orhan for their support and pleasant time.

I thank my parents, my sisters and brothers for the support they provided through my entire life. I felt their love in my heart whenever I needed.

Finally I would like to thank my one and only one Dilek, my beautiful wife, for her love, support and patience throughout my graduate study.

This research is supported by the Turkish State Planning Organization (DPT) under the TAM Project, number 2007K120610, and by the Bogazici University Research Fund under the grant agreement number “8684”.

ABSTRACT

USER, DEVICE, ORIENTATION AND POSITION INDEPENDENT HUMAN ACTIVITY RECOGNITION ON SMART PHONES

Smart phones equipped with a rich set of sensors are explored as alternative platforms for human activity recognition in the ubiquitous computing domain. However, there exist challenges that should be tackled before the successful acceptance of such systems by the masses. In this thesis, we particularly focus on the challenges arising from the differences in user behavior and in the hardware. To investigate the impact of these factors on the recognition accuracy, we collected data from 20 users focusing on five basic locomotion activities using the accelerometer, gyroscope and magnetometer. Using this dataset, we analyze whether activity recognition can be performed independently in terms of device, device model, user, device orientation and device position. We first show that, using raw acceleration, above 96% recognition accuracy can be obtained for device and model dependency tests, while success rate for orientation and user dependency tests remained at 87% and 90%. In order to tackle these issues, we first calculated linear acceleration, then using sensor fusion these acceleration readings are converted from phone coordinates to the earth coordinates. These methods helped in removing the orientation effects and increased both the user-independent and orientation-independent activity recognition accuracy to 98% and 95%. Finally, we analyze the impact of phone position on activity recognition using three different methods, namely using a generalized classifier, position-specific classifier and a joint classifier and show that using position-specific classification is not necessary, a generalized classifier performs very similarly. However, analyzing the confusion matrices, we observe that, stationary activities (sitting and standing) reduce the performance and combining these activities into a stationary class boosted recognition rates up to 98%.

ÖZET

AKILLI TELEFONLAR ÜZERİNDE KİŞİ, CİHAZ, CİHAZ YÖNÜ VE POZİSYONUNDAN BAĞIMSIZ EYLEM TANIMA

Zengin bir algılayıcı kümesi ile donatılmış akıllı telefonlar, insan eylemlerinin tanınmasında alternatif platformlar olarak yaygın hesaplama alanında araştırılmaktadır. Ancak bu tip sistemlerin kitleler tarafından başarılı bir şekilde kabul edilebilmesi için bir takım zorluklar mevcuttur. Bu tezde, bilhassa insan davranışlarından ve donanımdan kaynaklanan zorluklara yoğunlaşılmıştır. Bu faktörlerin eylem tanıma başarımı üzerindeki etkisini analiz edebilmek için, 20 katılımcıdan ivmeölçer algılayıcısı kullanılarak, beş temel hareket eylemi içeren bir veri kümesi toplanmıştır. Bu veri kümesi kullanılarak, eylem tanımanın cihaz, cihaz modeli, kullanıcı, cihaz yönelimi ve cihaz pozisyonu gibi telefon üzerinde eylem tanımayı etkileyecek faktörlerden bağımsız olarak yapılıp yapılamayacağı analiz edilmiştir. Öncelikle cihaz ve cihaz modelinden bağımsız bir şekilde eylem tanıma deneylerinde %96 başarım elde edilirken, cihaz pozisyonu ve kullanıcı bağımlılığı testlerinde %87 ve %90 başarım elde edilebildi. Bu konularla baş etmek için, öncelikle doğrusal ivmelenme değerleri hesaplandı ve ardından algılayıcıların birleşimi ile ivmenin telefon koordinatlarından dünya koordinatlarına çevrilmesi sağlandı. Bu metotlar ile, cihazın yöneliminin etkisi ortadan kaldırılarak, kişiden ve yönelimden bağımsız eylem tanımanın başarımı sırayla %98 ve %95'e yükselmiştir. Son olarak telefonun pozisyonun etkisi üç farklı metot, genel sınıflandırma, pozisyona-özel sınıflandırma ve ortak sınıflandırma kullanılarak analiz edilmiş ve pozisyona özel sınıflandırma yapmanın gerekli olmadığı genel sınıflandırma kullanmanın da benzer sonuçlar elde ettiği gösterilmiştir. Ancak, karışıklık matrislerine yakından bakıldığında sabit aktivitelerin (ayakta durma ve oturma) başarımı düşürdüğü ve bu aktiviteleri tek bir sınıfta birleştirilmesi ile eylemler %98 üzerinde bir başarım ile tanınabilmektedir.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS iii
ABSTRACT iv
ÖZET v
LIST OF FIGURES viii
LIST OF TABLES x
LIST OF SYMBOLS xiii
LIST OF ACRONYMS/ABBREVIATIONS xiv
1. INTRODUCTION 1
1.1. Contributions 5
1.2. Thesis Outline 6
2. BACKGROUND ON SENSORS AND ACTIVITY RECOGNITION PROCESS 8
2.1. Sensors 8
2.1.1. Accelerometers 8
2.1.2. Gyroscopes 9
2.1.3. Magnetic Field Sensors / Compass 10
2.2. Activity Recognition 11
2.2.1. Preprocessing 11
2.2.2. Segmentation 12
2.2.3. Feature Extraction 12
2.2.4. Dimensionality Reduction 12
2.2.5. Classification 13
3. RELATED WORK 14
3.1. Activity Recognition with Smartphone Sensors 14
3.2. Orientation-Free Activity Recognition 15
3.3. Position-Free Activity Recognition 16
3.4. Phone Position Recognition 22
4. DEVICE, ORIENTATION, USER AND POSITION INDEPENDENT ACTIVITY RECOGNITION 23

4.1. Data Collection: Raw Acceleration, Linear Acceleration and Coordinate Systems . . . 25
4.2. Segmentation . . . 32
4.3. Feature Selection: Simple and Extended Features . . . 32
4.4. Classification . . . 34
4.4.1. Activity and Position Classification . . . 35
5. PERFORMANCE EVALUATION . . . 38
5.1. Android Application . . . 38
5.2. Data Collection Scenarios . . . 38
5.2.1. Raw Acceleration Data Collection . . . 39
5.2.2. Linear Acceleration Data Collection . . . 40
5.2.3. Data Collection with Different Phone Positions . . . 40
5.2.4. Activity Recognition Phase . . . 41
5.3. Experimental Analysis Results . . . 41
5.3.1. Analysis of Device, Device-Model, Orientation, and User Dependency with Raw Acceleration Data . . . 41
5.3.2. Analysis of Orientation, and User Dependency with Linear Acceleration Data . . . 46
5.3.3. Analysis of Position Dependency on Activity Recognition . . . 49
5.3.4. Analysis of Position Recognition . . . 56
6. CONCLUSION . . . 60
APPENDIX A: FEATURE SELECTION DETAILS . . . 62
REFERENCES . . . 70

LIST OF FIGURES

Figure 1.1. Nexus 6 Smartphone equipped with a list of sensors. 2

Figure 2.1. Typical Steps of Activity Recognition Process [1]. 11

Figure 4.1. Activity and Phone Position Recognition Block. 24

Figure 4.2. Axes and rotation angles in Android OS [2]. 27

Figure 4.3. Conversion from Phone to Earth Coordinate System using accelerometer, gyroscope and magnetic field sensor. 31

Figure 4.4. The hourly battery impact (%) of the core sensors [3]. 31

Figure 4.5. K-Nearest Neighbors of a test vector x in a two dimensional vector space when k=10. 35

Figure 4.6. The position of smartphones in daily life [4]. 36

Figure 5.1. Activity recognition accuracy of position dependency tests using phone and earth coordinate systems for KNN. 50

Figure 5.2. Activity recognition accuracy of position dependency tests using phone and earth coordinate systems for DT. 51

Figure 5.3. Activity recognition accuracy of position dependency tests using phone and earth coordinate systems with KNN for 4 activities. . 55

Figure 5.4. Activity recognition accuracy of position dependency tests using phone and earth coordinate systems with DT for 4 activities. . . 55

Figure 5.5. Activity recognition accuracy of joint classification (Case 5). . . . 56

Figure 5.6. Activity recognition accuracy of joint classification with KNN (Case 5). 57

Figure 5.7. Activity recognition accuracy of joint classification with DT (Case 5). 57

Figure 5.8. Position recognition accuracy using phone and earth coordinate system for both classifiers. 58

Figure 5.9. Position recognition accuracy on activity basis using KNN and DT classifiers. 58

Figure 5.10. Average position recognition accuracy for each position. 59

LIST OF TABLES

Table 3.1. Detailed Comparison of Related Studies (OI:Orientation Independence, PI: Position Independence). . . . . .

Table 4.1. List of selected features for activity and position recognition, in each iteration of the tests. 33

Table 5.1. Comparison of orientation, model, device user dependency using overlapping and not overlapping windows and the classifiers. 42

Table 5.2. The effect of using different window sizes on recognition rates with the simple feature set. 43

Table 5.3. The effect of using an extended feature set and window sizes on recognition rate. 44

Table 5.4. Recognition rates after applying normalization on the feature set. . 44

Table 5.5. Recognition rates after applying filtering during the classification. . 45

Table 5.6. Confusion matrices for orientation tests with KNN and DT classifiers. 46

Table 5.7. Confusion matrices for user tests with KNN and DT classifiers. . . 46

Table 5.8. Accuracy of KNN and DT classifiers for different window sizes on phone and earth coordinate systems using linear acceleration. . . . 47

Table 5.9. Confusion matrices for orientation tests with KNN classifier using linear acceleration and different coordinate systems. 48

Table 5.10. Confusion matrices for user tests with KNN classifier using linear acceleration and different coordinate systems. 48

Table 5.11. Accuracy and F-Measure values for User and Orientation Tests. . . 49

Table 5.12. Confusion matrices of jacket position using phone and earth coordinates. 53

Table 5.13. Confusion matrices of backpack position using phone and earth coordinates. 54

Table 5.14. Confusion matrices of sidebag position using phone and earth coordinates. 54

Table A.1. Selection rates of features with 2.56 second windows and KNN classifier for raw acceleration experiments. 63

Table A.2. Selection rates of features with 2.56 second windows and DT classifier for raw acceleration experiments. 64

Table A.3. Selection rates of features with 2.56 second windows and KNN classifier for linear acceleration experiments. 65

Table A.4. Selection rates of features with 2.56 second windows and DT classifier for linear acceleration experiments. 66

Table A.5. Selection rates of features with 2.56 second windows and KNN classifier for position independent activity recognition experiments. . . 67

Table A.6. Selection rates of features with 2.56 second windows and DT classifier for position independent activity recognition experiments. . . 68

Table A.7. Selection rates of features with 2.56 second windows using KNN and DT classifiers for position recognition experiments. 69

LIST OF SYMBOLS

F_{c_i}	i^{th} FFT coefficient
$F_{c_{i-j}}$	i^{th} to j^{th} FFT coefficients
$I_{max_{corr}}$	Maximum correlation index of magnitude
T_m	Period of magnitude
zcr_m	Zero crossing rate of magnitude
μ_m	Mean of magnitude
μ_x	Mean of x axis
μ_y	Mean of y axis
μ_z	Mean of z axis
σ_m	Standard deviation of magnitude
σ_x	Standard deviation of x axis
σ_y	Standard deviation of y axis
σ_z	Standard deviation of z axis
σ_m^2	Variance of magnitude
σ_x^2	Variance of x axis
σ_y^2	Variance of y axis
σ_z^2	Variance of z axis

LIST OF ACRONYMS/ABBREVIATIONS

DT	Decision Tree
FFT	Fast Fourier Transform
HMM	Hidden Markov Model
KNN	K-Nearest Neighbor
OS	Operating System
SFS	Sequential Forward Selection
SVM	Support Vector Machine

1. INTRODUCTION

Human activity recognition systems using different sensing modalities, such as cameras or wearable inertial sensors, have been an active field of research in the domain of ubiquitous computing [5, 6]. Considering their potential to be applied in various application areas, including ambient assisted living, health and wellbeing monitoring, targeted advertisement, human activity recognition systems are becoming a part of our daily lives [7].

Recently, smart phones, equipped with a rich set of sensors, are explored as alternative platforms for human activity recognition [8–12]. Besides the inclusion of sensors, such as accelerometer, compass, gyroscope, proximity, light, GPS, microphone, camera, the ubiquity and the unobtrusiveness of the phones and the availability of different wireless interfaces, such as Wi-Fi, 3G and Bluetooth, make them an attractive platform for human activity recognition.

Figure 1.1 shows Google Nexus 6, as an example. It is equipped with an accelerometer, gyroscope, magnetometer, proximity sensor, ambient light sensor, haptics, hall effect, barometer sensors. Especially, motion sensors, such as accelerometer, gyroscope, magnetometer, are the mostly-utilized sensors for human activity recognition.

However, in spite of its advantages, human activity recognition on mobile phones also faces some challenges due to hardware limitations including battery, processing and storage constraints compared to more powerful stations. Due to the battery limitation, it is challenging to support continuous sensing applications as addressed in [13]. Considering the processing limitations, since human activity recognition requires running classification algorithms, originating from statistical machine learning techniques, it may be challenging to run resource-intensive classifiers on the phones [13, 14]. Other challenges stem from the different use of the phones by different people and the differences in the way people perform activities. For instance, the phone context problem [15] arises from the human behavior when the phone is carried in an inappropriate position



Figure 1.1. Nexus 6 Smartphone equipped with a list of sensors.

relative to the event being sensed. Especially with human activity recognition using inertial sensors, location where the phone is carried, such as in the pocket or in the bag, and the orientation of the phone affect the activity recognition performance.

Besides these challenges, since the classification of activities is mostly based on the use of statistical machine learning techniques, a learning phase is required. Mostly, supervised or semi-supervised learning techniques are utilized and such techniques rely on labeled data, i.e., associated with a specific class or activity. Labeling the data in the training phase is usually a tedious and complex process. In most of the cases, the user is required to label the activities and this, in turn, increases the burden on the user. Hence, user-independent training and activity recognition are required to foster the use of human activity recognition systems where the system can use the training data from other users in classifying the activities of a new subject. Besides the user-independent recognition, it is also important that human activity recognition algorithms should work on different mobile phone platforms in a device-independent manner for the acceptance of such systems by masses.

In a recent study [14], a review of activity recognition systems using mobile phones is provided. We observe that motion-based activity recognition, using inertial sensors is the dominating type of activity recognition on mobile phones, besides the systems using location-based activity recognition and motion-based activity recognition using wireless transceivers and other sensors, such as GPS. In the context of motion-based activity recognition systems, 3-axis accelerometers are the mostly utilized sensors available on the phones and most of the studies focus on detecting the locomotion activities, such as walking, standing, or transportation modes, biking, traveling with a vehicle.

In this thesis, we specifically focus on the challenges of user, device, orientation and phone-position-independent activity recognition on mobile phones to accelerate the acceptance of such systems in practical applications. In particular, we focus on the recognition of five basic locomotion activities, including running, walking, biking, standing and sitting, using fused data from the accelerometer, the magnetic field sensor (magnetometer) and the gyroscope available on most of the smart phone platforms. As the initial step, we collect data from twenty different users carrying different types of mobile phones at different orientations. First we process the data using simple features, including the mean, the variance and the standard deviation, where the features are extracted from the square sum of the three acceleration components on x, y and z axes and classify the activities using the K-Nearest Neighbors (KNN) and decision tree classifiers. Considering the results of this first set, we observed that although the activities can be recognized with a high accuracy in a user and device-independent way, the orientation of the mobile phone impacts the accuracy of the results and the accuracy is found to be around 83%. As the next step, we focused on the use of extended set of features, including FFT coefficients and autocorrelation based features, for orientation-independent activity recognition and this increased the accuracy to 85%. In the second group of tests, we focused on how to improve the performance further, and explored the use of linear acceleration, excluding the effect of gravitational force. With this modification, the accuracy was increased to 93%. One of the solutions discussed in the literature is to use signal transformation to overcome the orientation effects. Accordingly, as the final step, we focused on the use of earth coordinates in the classification process instead of using phone coordinates, which further improved

the classification performance to 97% accuracy.

After exploring the orientation-independence issue, we focus on the phone-position independence. The readings from motion sensors are sensitive to body position where the phone is carried. In this part, we explore the performance of activity recognition with three different methodologies: i) using a generalized classifier where the training phase includes data from all relevant positions, ii) using position-specific classifiers where a specific classifier is trained for each relevant position, iii) using a joint classifier, where the phone position and the activity is associated with a class, i.e., walking phone in the pocket and walking phone in the bag are different classes. We also explore the performance where training data from an individual is included or not included in the training data where testing the performance for an individual. In the literature, there are studies that report using a position-specific classifier does not bring any improvements compared to a generalized classifier [16,17], but there are also studies that report the opposite [18, 19]. Hence, there is no consensus on this and we also find that there is a very small improvement (2% on average), when a position-specific classifier is used compared to a generalized classifier. The effect of the location information on recognition performance is dependent on the activities and the positions involved. In [20], joint classification was reported to achieve higher accuracies compared to a generalized classifier and position-specific classifier. However, in our results joint classification exhibited worse performance than a generalized and position-specific classifier.

After examining the confusion matrices of the classifiers, we observe that identifying sitting and standing activities is very challenging in jacket pocket, backpack and side-bag positions, whereas in the pants pocket the accuracies are higher, as expected. Accordingly, we decided to combine these activities into a single "stationary" class. Using such a method, improved the average recognition accuracy by 2.8% for the pants, 20% for the jacket, 9% for the backpack, and 14% for the side-bag positions for a generalized classifier and similar increase is achieved for position-specific classification.

Although it is not the main topic of this thesis, we also looked at the performance of phone-position recognition using the same methodology. Besides the activity recog-

nition, knowing the phone position may have an effect on managing phone resources. For instance, if the phone is detected in the pocket, the volume of an incoming call automatically can be increased and vibrations can be turned on [21]. Using the same methodology in the activity recognition process, same set of features, same classification algorithms, we showed that phone position can be recognized with around 70% accuracy.

As we elaborate in Section 3, compared to the previous studies in the literature, although, a few of the previous works jointly analyze the orientation and position dependency [16, 19, 22] on the activity recognition accuracy, either they analyze the performance with only one method (using orientation-independent features, usually the magnitude or signal transformation), or they do not focus on user-independency and use cross-validation tests [16], or do not focus on how to estimate where the phone is carried. We follow a comprehensive approach and analyze the impact of each possible method and each possible parameter (device, model, user, orientation, position) on the performance including device and device-model dependency, which is usually ignored in the previous studies and the same models of devices are used.

1.1. Contributions

- We investigate the fundamental problems of activity recognition using smart phone sensors from several perspectives, including device and model (hardware) dependency, user dependency, orientation dependency and phone position dependency. We show that, while the recognition accuracy is not affected by the device and the device model, user, orientation and phone position dependency impact the results, when raw acceleration signals in phone coordinates are used.
- In order to provide orientation-independence, we show that using linear acceleration features, removing the effect of gravity, instead of raw acceleration features improves the results and signal transformation from phone coordinates to earth coordinates increases the accuracy even further.
- Considering the phone-position independence, we explore three different methods, including a generalized classifier, a position-specific classifier and a joint classifier,

and show that using a position specific classifier hardly improves the results. A generalized classifier also performs similarly.

- We collected physical activity data from 20 participants carrying three different phone models, at different orientations and four phones at the same time at four different positions. This data is made available for other researchers to be used as a benchmark in their studies.
- We investigate the impact of feature list on the activity recognition performance and by following a feature selection algorithm, we showed that using selected features exhibits better results. We also investigate the impact of the window size at the segmentation stage of activity recognition and show that using 2.56 or 5.12 seconds windows exhibit better performance compared to shorter window sizes.
- We also look at the phone position recognition problem using the same methodology as in the activity recognition process and showed that the position of the phone can be recognized with 70% accuracy on average using a user-independent leave-one-out approach.
- We analyze the performance of activity recognition by using different proposed methods in the literature to cope with orientation and position independence for a comparative evaluation on the same dataset.
- We showed that, in some positions it is very difficult to identify sitting and standing activities, except the pants case where the motion of the leg can be captured. Hence, combining these activities into a single activity, improves the results.

1.2. Thesis Outline

The rest of the thesis is organized as follows. In Chapter 2, we present some background information on the utilized smart phone sensors and the general activity recognition process using sensing information. In Chapter 3, we present the related work on activity recognition using mobile phone sensors, particularly those focusing on the impact of orientation and phone position on the activity recognition performance.

Chapter 4, introduces our methodology for activity recognition, the usage of linear acceleration, signal transformation from phone coordinates to earth coordinates, features list and the classifiers' details. In Chapter 5, we first explain the data collection experiments and then elaborate on the results of the experiments. Chapter 6 finally concludes the thesis and we discuss the future work therein.

2. BACKGROUND ON SENSORS AND ACTIVITY RECOGNITION PROCESS

In this section, we briefly give background information about the sensors used in this thesis and the steps of activity recognition process. First, we will explain the overall concepts like acceleration, orientation and magnetic field, with the sensors that are used for measuring these physical magnitudes. Then, the details of each step followed in the domain of activity recognition process will be given.

2.1. Sensors

2.1.1. Accelerometers

Acceleration can be defined as the rate of change in velocity with time. Acceleration of an object is determined by a net force applied to it with magnitude and direction. Since it is a vector quantity, any changes in the direction or speed is also called acceleration.

There are two kinds of acceleration: average and instantaneous acceleration. Average acceleration is measured over a finite time interval. In other words, it can be described as the difference in final and initial velocity on a predefined time interval. Instantaneous acceleration, in contrast, is the acceleration that is measured over infinitely small time interval. In calculus terms, it is idealized as the difference of initial and final velocity while the time interval is approaching to zero. Also there is a special case of acceleration called constant acceleration which combines the characteristics of average and instantaneous acceleration. Free fall is an example to constant acceleration caused by gravitational force where the both average and instantaneous acceleration is equal.

Acceleration of an object can be measured with the electromechanical sensor

devices called accelerometer. They are used for measuring the proper acceleration which is defined as acceleration relative to non-accelerating reference point. There are variety of usage areas such as medical applications, inertial navigation systems, structural monitoring, transportation systems and consumer electronics since they are capable of detecting tilt, vibration and orientation along with acceleration. In the domain of electronic systems, MEMs based accelerometer technology is widely used in smart phones. It enables device to detect physical nuances and sense the surrounding environment.

Most of the Android running devices are equipped with built-in acceleration sensors. Originally, accelerometers on the mobiles were included to enhance user experience by automatically changing the orientation of the display according to the orientation of the mobile phone held by the user. However, as widely adopted in the studies on activity recognition using body-worn sensors, accelerometers embedded in the mobiles can also be used to recognize the user's motion and motion-related activities. On Android devices, the accelerometer measures the acceleration force in m/s2 that is applied to a device on all three physical axes (x, y and z), including the force of gravity. Figure 2 shows the 3 axes of the accelerometer with respect to the device coordinates on an Android platform.

2.1.2. Gyroscopes

A gyroscope is a device that measures or maintains the orientation of an object, using the principles of angular momentum. Mechanically, a gyroscope is a spinning wheel or disc in which the axle is free to assume any orientation. Although this orientation does not remain fixed, it changes in response to an external torque much less and in a different direction than it would without the large angular momentum associated with the disc's high rate of spin and moment of inertia. The device's orientation remains nearly fixed, regardless of the mounting platform's motion, because mounting the device in a gimbal minimizes the external torque.

A free gyroscope maintains its axis. Gyroscopes can be used in combination with

compasses, which complement or replace magnetic compasses (in ships, aircraft and spacecraft, vehicles in general), to assist in stability or and inertial navigation systems. In addition to being used in compasses, gyroscopes have been introduced into consumer electronics. Since it allows the calculation of orientation and rotation, designers have incorporated them into modern technology. The integration of the gyroscope has allowed for more accurate recognition of movement within a 3D space than the previous lone accelerometer within a number of smart phones. Gyroscopes in consumer electronics are frequently combined with accelerometers (acceleration sensors) for more robust direction and motion sensing.

On Android platforms, gyroscope measures the rate or rotation in rad/s around a device's x, y and z axes. The coordinate system for the gyroscope is the same as the acceleration sensor's coordinates, as shown in Figure 2. Rotation is positive in the counter-clockwise direction. Similar to the accelerometer, it is included as a hardware sensor on Android.

2.1.3. Magnetic Field Sensors / Compass

A magnetic field is a mathematical description of the magnetic influence of electric currents and magnetic materials. The magnetic field at any given point is specified by both a direction and a magnitude (or strength); as such it is a vector field. The term is used for two distinct but closely related fields denoted by the symbols B and H. B refers to magnetic flux density, and H to magnetic field strength. Magnetic flux density is most commonly defined in terms of the Lorentz force it exerts on moving electric charges.

On Android platform, geomagnetic field sensor measures the geomagnetic field strength along the 3-axes. As a hardware-based position sensor, it is used for determining a device's physical position in the world's frame of reference [23]. It uses the same coordinate system as the gyroscope and the accelerometer.

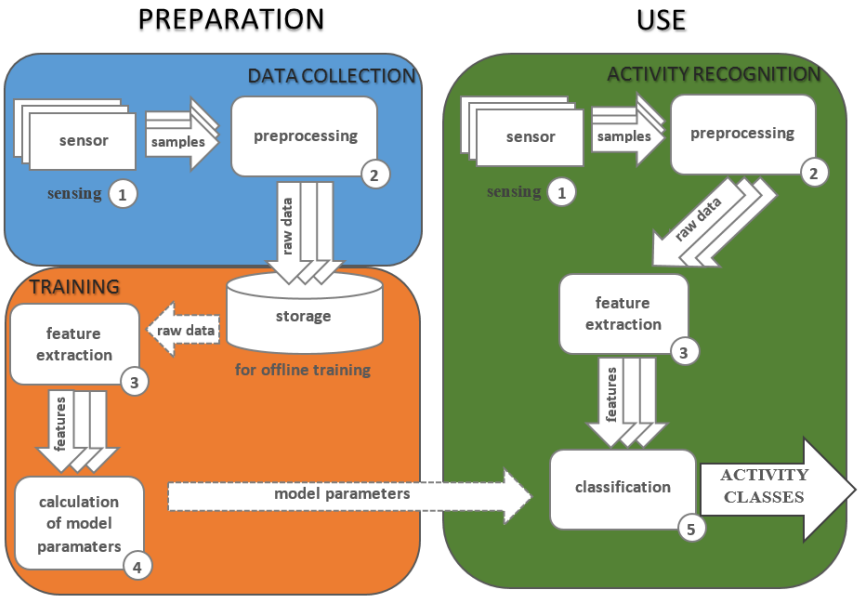


Figure 2.1. Typical Steps of Activity Recognition Process [1].

2.2. Activity Recognition

The activity recognition process can be summarized as determining a target set of activities, collecting sensor readings and assigning sensor readings to the appropriate activities. In other words, it is the process of how to interpret the raw sensor data to classify a set of activities. Many of the activity recognition studies, not necessarily in the field of mobile phone sensing, focus on the use of statistical machine learning techniques to infer information about the user activities from raw data [6,24]. Figure 2.1 presents the typical steps followed in the activity recognition process and we describe them in the following subsections.

2.2.1. Preprocessing

The preprocessing step in activity recognition contains filtering noisy output and removing irrelevant information generated from sensors by applying different kinds of operations such as low pass, high pass, Laplacian and Gaussian filters. Using these filters, data can be represented as smoother signals.

2.2.2. Segmentation

Segmentation is the process of dividing continuous data into meaningful pieces in an intelligent manner. For this purpose, different segmentation techniques like sliding windows, top-down and bottom-up can be applied on streaming data. These approaches have their own superiorities and weaknesses in comparison to each other. Sliding windows can be considered as the most popular and well-known technique in the literature. Basically, this technique is implemented as dividing streaming data into equal or varied sized pieces which can be also overlapping on miscellaneous ratios. Crucial part is to determine the optimal size and overlapping ratio of windows since these parameters have direct effect on the performance of classification. Using equal time windows may lead to decrease in the complexity of computation, make this approach preferable in the domain activity recognition.

2.2.3. Feature Extraction

Feature extraction involves simplifying the amount of resources required to describe a large set of data accurately. When performing analysis of complex data one of the major problems stems from the number of variables involved. Analysis with a large number of variables generally requires a large amount of memory and computation power or a classification algorithm which overfits the training sample and generalizes poorly to new samples. Feature extraction is a general term for methods of constructing combinations of the variables to get around these problems while still describing the data with sufficient accuracy.

2.2.4. Dimensionality Reduction

The dimensionality reduction is the process of reducing the number of random variables in meaningful approach. For high-dimensional datasets, dimension reduction is usually performed prior to applying classification algorithm. In order to reduce the dimensions of a feature set different methods are proposed which are grouped under two main approach. These are feature selection methods and feature transform methods.

The selection methods are based on selecting subset of elements from a main set whereas transform methods are used converting existing feature set into small highly delegated sets.

The main linear technique for dimensionality reduction, principal component analysis (PCA), performs a linear mapping of the data to a lower dimensional space in such a way that the variance of the data in the low-dimensional representation is maximized.

Feature extraction and dimension reduction can be combined in one step using principal component analysis (PCA), linear discriminant analysis (LDA), or canonical correlation analysis (CCA) techniques as a pre-processing step followed by clustering by KNN on feature vectors in reduced-dimension space.

2.2.5. Classification

The classification is the problem of identifying to which of a set of categories a new observation belongs, on the basis of a training set of data containing observations whose category membership is known. In the terminology of machine learning, classification is considered an instance of supervised learning, where a training set of correctly identified observations is available. The corresponding unsupervised procedure is known as clustering or cluster analysis, and involves grouping data into categories based on some measure of inherent similarity. There are several classification algorithms proposed in the domain of human activity classification. K-Nearest Neighbor, Decision Tree, Hidden Markov Models and Naive Bayes are most widely used classification algorithms in the literature.

3. RELATED WORK

In this section, we first review the related studies that focus on human activity recognition using smart phone sensors, in general. Next, we investigate the related work that specifically focus on orientation and position independent activity recognition. Finally, we review the studies on phone position/placement recognition.

3.1. Activity Recognition with Smartphone Sensors

In a recent study [14], a review of activity recognition systems using mobile phone sensors was provided. In general, location- and motion-associated activity recognition are the two dominating types of activity recognition using mobile phones. Location-based activity recognition systems aims at associating activities with places, and hence activities specific to a place can be recognized. For instance, in the Reality Mining project [25], three basic activities were targeted "home, work, elsewhere". In location-based activity recognition systems, usually, cell tower, WLAN, Bluetooth information is utilized besides GPS. However, the main disadvantage of such systems is that they can provide only coarse information on activity types, for instance being at work does not mean having a meeting or working in front of a computer or talking on the phone.

Another larger group of studies in the literature focus on motion-based activity recognition. In most of the studies, inertial sensors are utilized, whereas in some of them wireless transceivers (cellular, WiFi) [8, 9, 26] and in others, other sensors [10, 27] such as GPS, are used. In this section, we particularly review the studies that investigate activity recognition using inertial sensors as it is the main topic of this thesis.

Accelerometer is the mostly utilized sensor in the studies that make use of inertial sensors for activity recognition due to its potential in recognizing the phone movements and the orientation of the phone by using gravity information. Besides accelerometer, gyroscope together with accelerometer is also utilized in some of the related studies [28, 29]. Additionally, impact of using magnetometer is also investigated in some of the

related studies [18, 30]. In [31], a detailed analysis on the performance of these sensors in activity recognition, both when they are used alone, as well as in combination with each other, is provided.

There is a number of recent survey papers reviewing the literature in using inertial smart phone sensors for human activity recognition [1, 14, 32, 33]. In this section we specifically explore the studies that focus on orientation and position independence as a central theme in activity recognition.

3.2. Orientation-Free Activity Recognition

Orientation of the phone is an important parameter in the activity recognition process using inertial sensors, since the sensor readings are sensitive to orientation changes with accelerometer and gyroscope. If orientation-independence is not addressed, users are required to carry the phone in a specific orientation and this limits the practical usage of activity recognition. As mentioned in [1, 16], orientation-independence can be addressed using two fundamental methods. The first method is to use orientation-independent features. For instance, instead of computing features from individual 3-axis, features are calculated using the accelerometer magnitude compensating the effect of orientation changes in individual axes [34]. This method has been used in [10, 34, 35] by utilizing the accelerometer magnitude information.

The second method to encounter orientation changes is to use signal transformation. For example, the coordinate system of the phone is transformed into a global earth coordinate system [22]. This method has been used in [16, 29] by transforming the phone coordinate system to earth coordinate system. In [36], both of these approaches are used in combination. In this thesis, we also explore both of the approaches and find that transforming to earth coordinate system increases the recognition accuracy more than using only orientation-independent features. We also show that computing features from linear acceleration instead of raw acceleration improves the results for coping with the impact of orientation. A similar solution to our methodology was proposed in [37]. Gyroscope was used to detect the orientation of the mobile device. Also

in [38], a filtering approach was used to estimate and remove the gravity from each axis of the accelerometer. In our earlier work [39], we also analyzed the user, orientation, and device dependency on the performance of activity recognition using linear acceleration and coordinate transformation. However, analysis of position-dependency was ignored, only the KNN classifier was used, and the impact of segmentation and feature extraction were not investigated. Hence, the analysis was much limited.

3.3. Position-Free Activity Recognition

Position-independence, i.e. where the phone is carried, is also another important factor in activity recognition process. People carry their phones in different locations, such as in a bag, in a pocket, in hand while talking, and due to the differences in signals coming from the sensors at different positions it is a challenge to detect the user activities. In order to tackle this, different methods can be used. Similar to the orientation-independence solution, one of the methods is to use context-independent features. In this method, a generalized classifier is trained using data from all related positions and it has been used in [10, 16, 29].

Another method is to use different sub-classes for different body positions, such that each class is associated with an activity and a position. For instance, walking while the phone is in a pocket position and walking while the phone is carried in a bag are considered differently. This method has been investigated in [20, 22, 40]. In [40], ActiServ, a service-based recognition architecture, is presented. ActiServ aims at minimal personalization effort by the users, and it is mentioned that only 1-3 minutes of training data for each activity is adequate. The performance of the system is tested with 20 users and from each user 2-3 minutes of data was collected using the OpenMoko Neo Freerunner phone. It is reported that if position specific training is used and the system is tested at the same position as in the training data, the performance of the system is observed to be 97.3% in terms of accuracy. However, if the training data and testing data are collected from different positions, the accuracy descents to 60%.

Another method used for position-independent activity recognition is to train a

separate classifier for each position. This method has been investigated in [16–19, 41]. In [17], the focus is only on the walking activity and other investigate the impact of position on walking speed estimation. It is concluded that, using position-specific training hardly improves the results. Similarly in [16], it is shown that using position-specific training improves the results 1-2% on average. In this study, 6 activities, were targeted, lying, sitting, standing, walking, running, and jumping, and 3 positions were tested, pocket (shirt), waist (belt) and trousers pocket. Different than these studies, in [18], it is demonstrated that in certain settings a priori information on location enhances activity classifier accuracy. Here, the position set was larger: front and back trouser pockets, in the shirt and jacket pockets, in a short-strap and long-strap bag, in a backpack, in an armband and in a waist. The activity set consisted of slow, normal and rush walking, and running. Similarly in [19], it is reported that using position-specific classification improves the recognition accuracy by 3% on average, compared to a generalized classifier.

In [20], these three methods are compared where the focus is on patient mobility, such as Parkinson patients, using smart phone sensors. 3 basic activities are targeted: sit, stand and walk. However, sit-to-stand and stand-to-sit were included as different activities to detect the transitions. 4 different phone positions were considered: hand, belt, pocket and bag. Using HMM and SVM classifiers authors investigate the performance by using a generalized classifier, where no position information is used, a position-specific model where the position information is known and a joint classifier where for each position and activity, there was a different class. Collecting data from 12-healthy participants, they show that a joint classifier performs better than the other approaches. Similar to this study, we also investigate these 3 methods in this thesis for a comparative evaluation. However, we also have 2 more cases where person-specific training was used and we target other types of activities and phone positions. Different than [20], in our case, the joint classifier performs much worse than the generalized classifier and the position-specific classifier.

Data collection experiments performed in [16, 18, 20] can be considered similar to our study. However, in [18] device and device-model tests were ignored and the

impact of the orientation was not considered. In [20], similarly the focus was on position dependence and orientation, device and model dependencies were not considered. In [16], both orientation and position dependency were considered. However, for orientation dependency, only the impact of using signal transformation was considered and orientation-independent features were ignored and for position dependency, a generalized classifier and a position-specific classifier were considered and a joint classifier was ignored. Additionally, device and model dependency were not the focus. Hence, our main contribution is to explore the parameter space for device, model, orientation, position and user-independent activity recognition by analyzing the performance with different possible methods. We follow a comprehensive approach and analyze the impact of each possible method and each possible parameter.

As reported, in the literature there are studies which state that position-specific activity recognition improves the results [18, 22]. In order to perform position-specific recognition, first the phone position should be estimated and then a classifier built for this specific position should be used. Phone position recognition has been previously addressed in [17, 18, 21, 42], usually using multiple sensors. In [18], light, proximity, magnetometer, gyroscope, gravity, acceleration and linear acceleration sensors were employed. In this study, the position recognition accuracy was reported as 92.94% with the best set of all features and 66% with a selected set of features. In [21], the focus was on in-pocket versus out-of pocket detection and using only the microphone, 80% accuracy was reported to detect these two classes. The use of accelerometer for position recognition was discussed in [17], where bag, ear, hand and pocket positions were considered. The average accuracy was reported as 94%, however, the activity set included only walking. In [20], phone position recognition was evaluated with three different method which are generalized classifier, activity-specific classifier and joint classifier using accelerometer. It was shown that the best results are achieved with the walking activity. However, the performance of each method varies with the position and the activity. In this thesis, we explore all these solutions on the same dataset aiming at a comparative evaluation. We conclude that, position-specific training does not improve the results compared to a general classifier. Moreover, joint classification performs worse than the other methods on our dataset.

In Table 3.1, we provide a taxonomy on the related studies that focus on orientation and position independence. This table also provides a comparative table of this thesis in relation to the related studies. Although, a few of the works jointly analyze the orientation and position dependency [16, 19, 22] on the activity recognition accuracy, either they analyze the performance with only one method (using orientation-independent features, usually the magnitude or signal transformation), or they do not focus on user-independency and use cross-validation tests [16], or do not focus on how to estimate where the phone is carried. We follow a comprehensive approach and analyze the impact of each possible method and each possible parameter (device, model, user, orientation, position) on the performance including device and device-model dependency, which is usually ignored in the previous studies and the same models of devices are used.

Table 3.1. Detailed Comparison of Related Studies (OI:Orientation Independence, PI: Position Independence).

Reference	**Activities**	**Phone Type**	**Classifiers**	**OI vs. PI**	**Positions**	**OI Method**	**PI Method**
Reddy *et al.*, [**10**]	Still, walk, run, bike, motorized	Nokia N95, iPhone, TMobile G1	Decision Tree, DHMM	OI	Arm, bag, chest, hand, pocket, waist	OI Features	PI- Generalized classifier
Siirtola *et al.*, [**35**]	Idling, walking, cycling , driving, running	Nokia N8, Samsung Galaxy Mini	QDA	OI	-	OI Features	-
Siirtola *et al.*, [**34**]	Walking, running, cycling, sitting/standing, driving a car	Nokia N8	Decision Tree	OI	Trouser's pocket, jacket's pocket, backpack, ear , brachium, table	OI Features	PI - Generalized classifier
Lu *et al.*, [**22**]	Walking, running, still, biking, in vehicle	Nokia N95, iPhone	Decision Tree	OI and PI	Front and back pants pocket, hand, jacket pocket, backpack, belt	OI - Signal transformation	PI – Position and activity joint classifier
Anjum *et al.*, [**29**]	Walking, running, walking upstairs, walking downstairs, still, biking, driving a car	Samsung Galaxy W	Decision Tree	OI	Pant, pocket, hand, hand bag, shirt pocket	OI - Signal transformation	PI - Generalized classifier
Berchtold *et al.*, [**40**]	Phone in hand, typing text messages. talking on phone, walking, standing, sitting, walking upstairs, biking, laying, phone on table/detached	OpenMoko Neo Freerunner	Fuzzy Classification	PI	Table, hand, pants pocket	-	PI – Position and activity joint classifier

(cont.)

Table 3.1. Detailed Comparison of Related Studies (OI:Orientation Independence, PI: Position Independence).

(cont.)

Reference	**Activities**	**Phone Type**	**Classifiers**	**OI vs. PI**	**Positions**	**OI Method**	**PI Method**
Thiemjarus *et al.*, [**16**]	Lying, sitting, standing, walking, running, jumping	iPhone	KNN	OI and PI	Pocket (L1), waist (L2) and trouser	OI - Signal transformation	PI - Generalized classifier, position-specific classifier
Antos *et al.*, [**20**]	Sit to stand, sitting, stand to sit, standing, walking	T-Mobile G1	HMM and SVM	PI	Bag, belt, hand, pocket	-	PI – Position and activity joint classifier
Park *et al.*, [**17**]	Walking	Nokia N900	Decision Tree and SVM	PI	Bag, ear, hand, pocket	-	PI – Generalized classifier, position-specific classifier
Martin *et al.*, [**18**]	Slow, normal and rush walking and running	Nexus S	Decision Tree, Decision Table, Naïve Bayes	PI	Front and back trouser pockets, in the shirt and jacket pockets, in a short-strap and long-strap bag, in a backpack, in an armband and in a waist	-	Generalized classifier, position-specific classifier
Sun *et al.*, [**19**]	Stationary, walking, running, bicycling, ascending stairs, descending stairs and driving	Nokia N97	SVM	OI and PI	6 pockets (2 front, 2 rear trousers pockets and 2 front jacket pockets) 4 orientations inside the pocket	OI Features	Generalized and position-specific classifier
This thesis	Sit, stand, walk, run, bike	Samsung Galaxy W, Samsung Galaxy S3	Decision Tree and KNN	OI and PI	Trousers pocket, jacket pocket, backpack, side bag	OI features and signal transformation	Generalized classifier, position specific classifier, location and joint classifier

3.4. Phone Position Recognition

As reported, in the literature there are studies which state that position-specific activity recognition improves the results [18, 22]. In order to perform position-specific recognition, first the phone position should be estimated and then a classifier built for this specific position should be used. Phone position recognition has been previously addressed in [17, 18, 21, 42], usually using multiple sensors. In [18], light, proximity, magnetometer, gyroscope, gravity, acceleration and linear acceleration sensors were employed. In this study, the position recognition accuracy was reported as 92.94% with the best set of all features and 66% with a selected set of features. In [21], the focus was on in-pocket versus out-of pocket detection and using only the microphone, 80% accuracy was reported to detect these two classes. The use of accelerometer for position recognition was discussed in [17], where bag, ear, hand and pocket positions were considered. The average accuracy was reported as 94%, however, the activity set included only walking. In [20], the evaluation of phone position recognition was performed using three different methods which are generalized classifier, activity-specific classifier and joint classifier using accelerometer. The highest accuracy is obtained with the walking activity. However, each method performed divergent on position and activity basis.

Although, it is not the main focus of this thesis, we also analyze the performance of phone position recognition using the same settings used for the activity recognition, linear acceleration, same set of features and classifiers. Approximately 70% accuracy is achieved and the performance of phone position recognition varies according to the activity and the type of the acceleration data, i.e., using phone or earth coordinates.

4. DEVICE, ORIENTATION, USER AND POSITION INDEPENDENT ACTIVITY RECOGNITION

In this chapter, we explain the methodology used in this thesis for coping with the impact of the device, orientation, user and position/context dependency on the activity recognition process. With respect to the steps of activity recognition explained in Chapter 2 and Figure 2.1, we explain the methods utilized at each step.

The objective of our study is to investigate a possible solution for activity recognition on smart phones that will enable classification of five different daily activities with high accuracy, providing not only independency of orientation but also device/model and user independency, as well. The activity recognition can be simply defined as the process of how to interpret the raw sensor data to classify a set of activities [14]. As mentioned, in the literature, statistical machine learning techniques are used to infer information about the activities from raw sensor readings and this process usually includes a training phase and a testing phase. The training phase requires the collection of labeled data to learn the model parameters. Once the data is collected, preprocessing, segmentation, feature extraction and classification steps follow to build the training model.

Although activity recognition is the main focus in this thesis, we also investigated the phone position recognition problem. In order to analyze the impact of position dependency on the activity recognition performance, we collected data from 4 different phone positions: pants pocket, backpack, side bag and jacket's pocket. One of the methods that we analyzed was to use position-specific activity classification. However, in order to perform position-specific classification, we first need to identify the phone position. In our experiments, we investigated how accurately we can perform the phone position recognition for position-specific activity classification. Besides activity recognition, position recognition can be used for managing the resources of the phone. As an example, if the phone is detected in the pocket, the phone screen can be automatically

locked [21].

Figure 4.1 shows the block diagram of activity and phone position recognition steps. In the data collection step, embedded sensors are used to compute gravity component in order to obtain dynamic acceleration value. This value can either be read in phone coordinate system or converted into earth coordinate system with the help of magnetic field sensor. Afterwards, time and frequency domain features are calculated for each segment of the data. In the final step, the classifier algorithm is used for the recognition of activities. In the following subsections we explain the steps of activity and phone position recognition followed in this thesis.

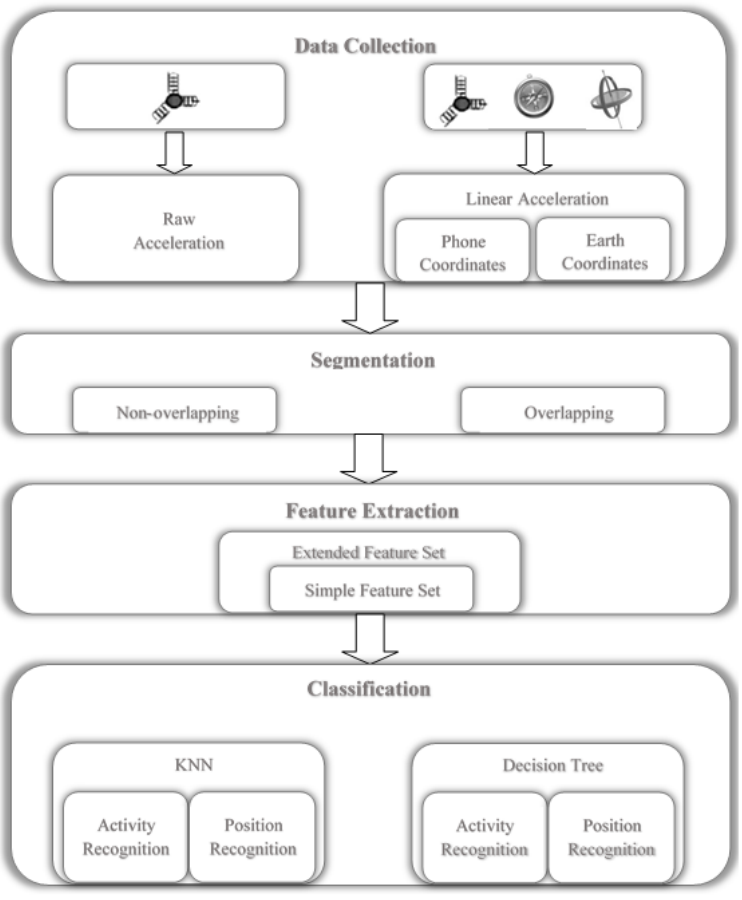


Figure 4.1. Activity and Phone Position Recognition Block.

4.1. Data Collection: Raw Acceleration, Linear Acceleration and Coordinate Systems

The embedded accelerometer of a smart phone measures the acceleration applying on the phone. If the device is not subject to any other acceleration, such that the device is stationary or moves with constant speed, the accelerometer only measures the gravity that points toward the center of the Earth. However, if there exists acceleration other than gravity, phone will measure the combination of both this acceleration and the gravity. In our work, we refer to this as "raw acceleration". In general, the raw acceleration can be defined as follows:

$$\text{Raw Acceleration} = \text{Linear Acceleration} + \text{Gravity}$$

Linear acceleration can be defined as the acceleration without the effect of gravity. It can be calculated by the subtraction of the gravity components on x, y and z axes from the same three axes with the raw accelerometer values. The gravity component measured on each axis of an accelerometer changes according to the orientation of the phone. If the orientation can be calculated correctly, it would be possible to find the corresponding gravity vector being applied on each axis respectively. Android OS sensor framework uses accelerometer, gyroscope and magnetic field sensors to obtain the linear acceleration and the gravity components on each axis separately.

These sensors have strengths and weaknesses but by using sensor fusion their weak points can be alleviated. An accelerometer measures gravity but it can fail under any other linear acceleration and also the output is usually noisy. Gyroscope measures angular velocity, the output is responsive but it adds drifts to output which causes measurement errors over time. Magnetometers provide heading towards the magnetic field of earth but they are easily effected from other magnetic fields and also their response time is low which can cause erroneous measurements on sudden movements. Gyroscope can measure angular speed on 3-axis, by integrating these outputs, absolute orientation of the phone could be obtained. But small errors in

the output of gyroscope can add noticeable drift with the integration over time. The orientation of the smartphone also could be determined by using the accelerometer and the magnetic field sensor pair. If the device is not subject to any other acceleration (the device is stationary or moves with constant speed), the accelerometer measures the gravity acceleration that points toward the center of the Earth, so with this information, the tilt angles around x and y axis can be determined. Also, the magnetic field sensor provides the magnetic vector in three axes. This sensor could be utilized to derive the device's z axis rotation angle which is defined as the angle between the magnetic north direction and the y axis. With the fusion of the data from the accelerometer and the magnetic field sensor, it is possible to detect the orientation of the device. But this combination has some drawbacks like noisy output from accelerometer or low response time and inaccurate readings from the magnetic field sensor. In order to overcome these limitations, Android sensor framework integrates the gyroscope into the system. The gyroscope has high response time and provides smooth outputs but its frailness comes from the fact that it drifts over time. The gyroscope provides rotation speeds relative to the phone's own coordinate system and this data can be used to correct errors caused by the accelerometer and the magnetic field sensor. With the fusion of these sensors, using the rotation vector which contains the rotation angles along each axis, it is possible to calculate the orientation of the phone. So, the gravitational force being applied on each axis of the accelerometer can be excluded and pure accelerometer values (linear acceleration) can be obtained.

Accelerometer and the magnetic field sensor can also be utilized to convert the acceleration output of the smart phone from its body frame of reference to the earth frame of reference. With the capabilities of accelerometer and magnetometer, if all acceleration outputs are generated according to a single reference point, similar measurements can be observed for the same set of activities regardless of the orientation of the phone. But in order to do that, we should cover some basic concepts about the coordinate systems, orientation and rotation. Figure 4.2 shows the 3 axes phone coordinate system in general.

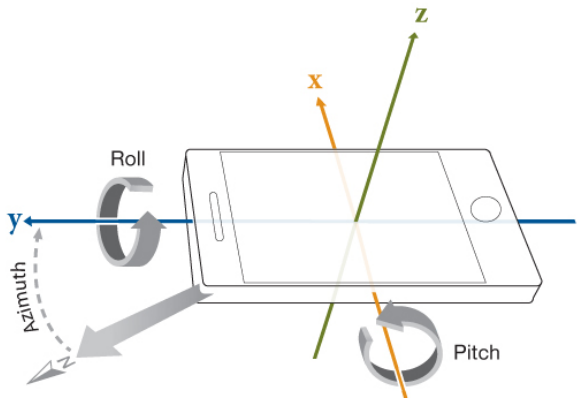


Figure 4.2. Axes and rotation angles in Android OS [2].

When a smartphone is put on a table, the x axis is horizontal and points to the right, y axis is vertical and point to the up, and z axis points out of the screen. This coordinate system is independent from the orientation of the phone. In order to express accelerometer readings in terms of the Earth's coordinate system rather than the phone coordinate system, we need to describe angular changes in rotation using rotation angles which are named as Azimuth, Pitch and Roll. The order in which these angles are applied to determine the orientation is important. Because applying these angles in a different order may lead to completely different final orientations. The accepted convention on applying these angles are first azimuth then pitch and finally roll.

- Azimuth "Ψ" rotation around z axis and its range is from 0^0 to 360^0.
- Pitch "Θ" specifies rotation around x axis and its range is from -180^0 to 180^0.
- Roll "Φ" describes rotation around y axis axis and its range is from -90^0 to 90^0.

Pitch and roll angles can be determined using the gravity measuring characteristic of the accelerometer. Azimuth angle can be computed by utilizing the magnetic field sensor as a compass. A three-axis accelerometer-embedded smartphone which is put horizontally on a table, can measure the gravitational force $\mathbf{g}$, in Earth's coordinate system with Equation 4.1:

$$\mathbf{G}_p = \begin{pmatrix} G_{px} \\ G_{py} \\ G_{pz} \end{pmatrix} = \mathbf{Rg} = \mathbf{R} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \tag{4.1}$$

$\mathbf{G_p}$ represents the magnitude of gravity vector measured in all three axis by $\mathbf{G_{px}}$, $\mathbf{G_{py}}$ and $\mathbf{G_{pz}}$ respectively. Since the device is placed horizontally on the table, accelerometer measures the gravity only on z axis. $\mathbf{R}$ is the rotation matrix describing the orientation of the phone relative to the Earth's reference of frame. Rotation matrix R can be composed of pitch, roll and azimuth rotation matrices. The orientation of the smartphone can be calculated by using these azimuth, pitch and roll rotations from a known initial position. Also pitch, roll and yaw angles can be expressed in terms of rotation matrices as follows:

$$\mathbf{R}_x(\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & cos(\theta) & sin(\theta) \\ 0 & -sin(\theta) & cos(\theta) \end{pmatrix} \tag{4.2}$$

$$\mathbf{R}_y(\phi) = \begin{pmatrix} cos(\phi) & 0 & -sin(\phi) \\ 0 & 1 & 0 \\ sin(\phi) & 0 & cos(\phi) \end{pmatrix} \tag{4.3}$$

$$\mathbf{R}_z(\psi) = \begin{pmatrix} cos(\psi) & sin(\psi) & 0 \\ -sin(\psi) & cos(\psi) & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{4.4}$$

$$\mathbf{R}_{xyz}\begin{pmatrix}0\\0\\1\end{pmatrix} = \mathbf{R}_x(\theta)\mathbf{R}_y(\phi)\mathbf{R}_z(\psi)\begin{pmatrix}0\\0\\1\end{pmatrix} \tag{4.5}$$

With the multiplication of rotation matrices in Equations 4.2, 4.3 and 4.4 we get the following rotation matrix in Equation 4.6:

$$= \begin{pmatrix}-sin(\phi)\\cos(\phi)sin(\theta)\\cos(\phi)cos(\theta)\end{pmatrix} \tag{4.6}$$

$$\frac{\mathbf{G}_p}{\|\mathbf{G}_p\|} = \begin{pmatrix}-sin(\phi)\\cos(\phi)sin(\theta)\\cos(\phi)cos(\theta)\end{pmatrix} \Rightarrow \frac{1}{\sqrt{G_{px}^2+G_{py}^2+G_{pz}^2}}\begin{pmatrix}G_{px}\\G_{py}\\G_{pz}\end{pmatrix} = \begin{pmatrix}-sin(\phi)\\cos(\phi)sin(\theta)\\cos(\phi)cos(\theta)\end{pmatrix} \tag{4.7}$$

Equation 4.6 can be expressed as Equation 4.7 to extract pitch and roll angles from the normalized accelerometer reading G_p. Using Equation 4.7, pitch and roll angles can be calculated as follows:

$$tan(\theta)_{xyz} = (\frac{G_{py}}{G_{pz}}) \tag{4.8}$$

$$tan(\phi)_{xyz} = (\frac{-G_{px}}{G_{py}sin(\theta) + G_{pz}cos(\theta)}) = \frac{-G_{px}}{\sqrt{G_{py}^2 + G_{pz}^2}} \tag{4.9}$$

2-axis magnetic field sensors can measure the azimuth in horizontal plane accurately if it is flat to the magnetic field of earth. The changes in the inclination of magnetic field sensor will introduce errors to the calculation of azimuth. So tilt compensation is needed to remove erroneous output. Smartphones are equipped with 3-axis magnetometers and with tilt compensation feature, azimuth angle can be calculated as:

$$tan(\psi) = (\frac{Y_{horizontal}}{X_{horizontal}}) \tag{4.10}$$

X and **Y** refer to the horizontal readings from the magnetic field sensor. With mathematical origins of the azimuth, pitch and roll calculation are shown, rotation angle calculation can be done easily. Android OS defines the rotation matrix as a virtual sensor like any other real sensors, accelerometer, gyroscope or magnetic field sensor. Sensor manager generates corresponding rotation matrix according to the provided gravity and magnetic field outputs. Hence, linear acceleration readings in phone coordinates A_{Lp} can easily be converted to the reference frame of earth A_{Le} with the following formula which is similar to the Equation 4.1:

$$\mathbf{A}_{Le} = \mathbf{R}.\mathbf{A}_{Lp} \tag{4.11}$$

As illustrated in Figure 4.3, using the capabilities of the three sensors, the coordinate system of the phone could be converted to the earth magnetic coordinate system [23]. So fused sensor system generates the same acceleration output independent of orientation of the phone during the motion. In our data collection experiments, the rotation matrix provided by the Android system was used to convert the linear acceleration readings from the phone coordinates to the earth coordinates using Equation 4.11.

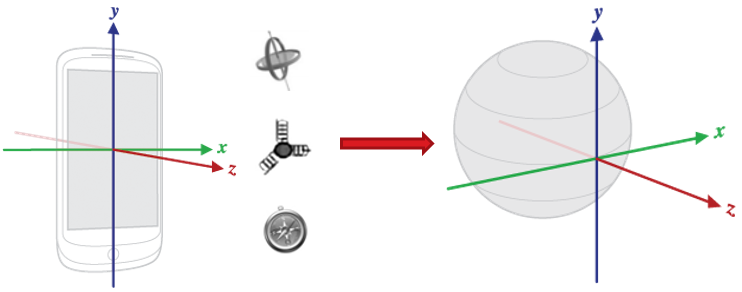


Figure 4.3. Conversion from Phone to Earth Coordinate System using accelerometer, gyroscope and magnetic field sensor.

Using the fused sensor output increases battery usage as well. In a related study [3], using accelerometer, gyroscope and magnetometer at the same, depletes battery 4 times faster compared to using only accelerometer. But since all these sensors have small impact on battery usage, this effect is acceptable in terms of battery life.

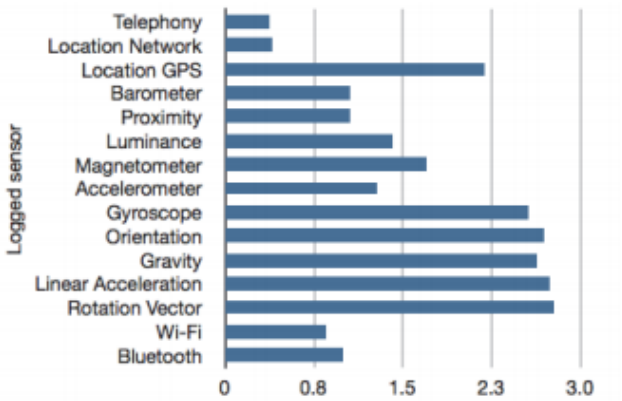


Figure 4.4. The hourly battery impact (%) of the core sensors [3].

4.2. Segmentation

Segmentation could be stated as the first preprocessing step of the activity recognition block. Selecting ideal size for data partitions is vital on the success of the activity recognition process. We used equal length non-overlapping and overlapping windows in our research since they are proven to be lightweight in terms of computational needs. In the first step, we examined the effect of using non-overlapping and overlapping windows on the recognition accuracy. With the 50% overlapping ratio, we obtained significant increase in the recognition results for both KNN and DT classifiers. So we decided to use overlapping windows in the following steps. Then we investigated the effect of window size on the classification performance. We again compared classification accuracies of KNN and Decision Tree algorithms using simple and extended feature sets segmented on different lengths. Some of the entities in the extended feature set were calculated using FFT. The FFT based calculations need the length of data to be power of 2, if not zero padding is applied to fit the length for calculation. In order to avoid zero padding, we selected data lengths as 1.28, 2.56 and 5.12 seconds and computed features of simple and extended sets according to these values. We did not use the window lengths under 1.28 second because it would not be possible to distinguish periodic activities under this time span.

4.3. Feature Selection: Simple and Extended Features

In the feature extraction step, first, we calculated time domain features like mean, variance and standard deviation from the magnitude of accelerometer axes, and also each individual axis separately. With the low accuracy results encountered in the orientation and user independent experiments, we decided to supplement the feature set with the frequency based features like zero crossing rate, maximum correlation index, period and FFT coefficients. Frequency based features need more computation effort but they are good at detecting periodic patterns in the activity data. As indicated in Figure 4.1, extended feature set consists of both time and frequency based features while simple feature set consists only time domain features.

After the calculation of simple and extended feature sets, dimensionality reduction procedure was applied with both classifiers to obtain the optimal combination of features that will be used as system parameters on the construction of activity recognition block. As introduced in Chapter 2, there are several techniques for dimensionality reduction but in this research, we used feature selection approach instead of the feature transformation techniques.

In this research, sequential forward selection (SFS) algorithm is used as the feature selection technique. Sequential forward selection is simply based on creating candidate feature subsets of the main set and adding each element, that increases the prediction accuracy, one by one to the feature set until there is no further improvement [43]. In our study, SFS algorithm is executed 10 times for each case to avoid a local maxima and minima. In the each iteration of experiments, feature selection algorithm was executed for both classifiers to obtain best fitting feature set. The selection rates of features on experiment basis are presented in Tables A.1, A.2, A.3, A.4, A.5, A.6 and A.7 in Appendix A. Table 4.1 summarizes the selected feature sets in the each batch of experiments.

Table 4.1. List of selected features for activity and position recognition, in each iteration of the tests.

Iteration	KNN	DT
Raw Acceleration	$\sigma_m, \mu_z, \sigma_z, zcr_m, I_{max_{corr}}, T_m, F_{c_{1-10}}$	$\sigma_m, \mu_z, \sigma_z, zcr_m, I_{max_{corr}}, T_m, F_{c_{1-10}}$
Linear Acceleration	$\mu_m, \sigma_m, \mu_x, \mu_y, \mu_z, \sigma_z, zcr_m, I_{max_{corr}}, T_m, F_{c_{1-2}}$	$\mu_m, \sigma_m, \sigma_x^2, \sigma_y^2 \mu_z, \sigma_z, zcr_m, I_{max_{corr}}, T_m, F_{c_{1-2}}$
Position Tests	$\mu_m, \sigma_m, \mu_x, \mu_y, \mu_z, \sigma_z, zcr_m, I_{max_{corr}}, T_m, F_{c_{1-10}}$	$\mu_m, \sigma_m, \sigma_x^2, \sigma_y^2, \mu_z, \sigma_z, zcr_m, I_{max_{corr}}, T_m, F_{c_{1-6}}, F_{c_{8-10}}$
Position Recognition	$\mu_m, \sigma_m, \mu_x, \sigma_x, \mu_y, \sigma_y, \mu_z, \sigma_z, zcr_m, I_{max_{corr}}, T_m, F_{c_{2-3}}, F_{c_{5-8}}, F_{c_{10}}$	$\mu_m, \sigma_m, \mu_x, \sigma_x, \mu_y, \sigma_y^2, \sigma_y, \mu_z, \sigma_z, zcr_m, I_{max_{corr}}, T_m, F_{c_{1-5}}, F_{c_{7-8}}, F_{c_{10}}$

4.4. Classification

As outlined in Section 1, in this study we focused on two different domains in smartphone based pattern recognition studies. Primarily, we probed activity recognition on smartphones phenomenon from several perspectives. As a secondary and minor branch, we have researched position classification using same sensors with activity recognition study. In order to do that, we selected K-Nearest Neighbors and Decision Tree algorithms for performance evaluation since they are both lightweight and need less computational resource compared to more complex algorithms in the literature. Moreover, they are widely used in the litearture [1,14,35,44]. These conditions make both algorithms perfect candidates for classification using smartphone.

K-Nearest Neighbors. In pattern recognition, K-Nearest Neighbors is a type of instance-based learning algorithm and it is used for classification and regression. In a multidimensional space, given N training vectors, KNN algorithm identifies the K nearest neighbors of any test sample vector using a distance metric. Basically, label of the test vector V can be estimated by calculating the distance to each training vector in terms of a distance metric. The class label of the test vector is determined by selecting the frequent class label in the K nearest neighbor sample set. There are several distance metrics used in the algorithm such as Euclidean, Minkowski and Mahalanovis distance. In our research, we used Euclidean distance as distance metric.

Figure 4.5 shows the classification of vector x using Euclidean distance in a two dimensional vector space consists of class labels + and -. The nearest K neighbor set of vector x consists of three + class and seven - when K is selected as 10, as a result the class label is predicted as -.

Decision Tree. Decision Tree is a supervised learning method used in machine learning, artificial intelligence, statistics and data mining. Decision Tree, in machine learning, is a prediction model that estimates the value of a variable, using a previously constructed tree based on the given observations. When the set of values that the target variable can take is finite, it is called classification tree. In classification trees,

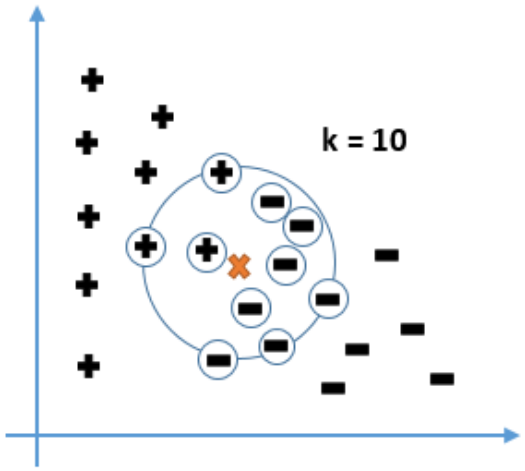


Figure 4.5. K-Nearest Neighbors of a test vector x in a two dimensional vector space when k=10.

branches represent features and leaves represent class labels. A decision tree can be built by splitting training data set into the smaller subsets using the values of the features. In this phase, the process of progressively splitting the data into small subsets is called recursive partitioning. Recursive partitioning continues until there is no further labelling in the branches of a node. After the training phase, classification begins at the root node and follows the path to the corresponding leaf node. Decision trees are easy to understand and visualize, performs well and fast even with large training sets.

4.4.1. Activity and Position Classification

As a major part of our study, we have studied the classification of daily activities including walking, running, biking, sitting and standing using embedded sensors of smartphones. The selection of this set of activities was motivated by the popularity of these classes in the related studies as also identified in the survey studies [1,14,32,33], so that we can compare our findings with similar studies. We evaluated the performance of the activity classification according to the several dependency conditions:

Orientation. Most people carry their smartphones on their outfit, trousers, jackets, shirts and sometimes on their bag. For example, when the phone is placed in

the pocket of trousers, it usually stays in the vertical orientation because of the tight environment. But when it is placed in the side pocket of a jacket, it is tend to stay in horizontal state. Or there may be orientation transitions according to loose carrying environments. The differences in orientation of the phone directly effects the classification accuracy since the readings from sensor outputs may vary.

Phone model. There are several phone models in the market and every model have different capabilities or sensor sets. Also there are many sensor models that is produced by different semiconductor manufacturers. The model diversity in integrated chip models and calibration problems even in the same chip models result in different readings on the sensor outputs. All these conditions effect the accuracy of the activity classification in smartphones.

User. People have different movement characteristics that arise from their physical conditions. The movement patterns of each person are determined directly from their postures and gaits. When creating an activity classification model, selecting a wide training set is important on accurate predictions of the classifier but most of the time this is not possible. So selecting the correct features that highlights the characteristics of different activities play important role on decreasing user dependency of classification.

Position. The position/placement of the phone while performing an activity impacts the readings of the sensor.

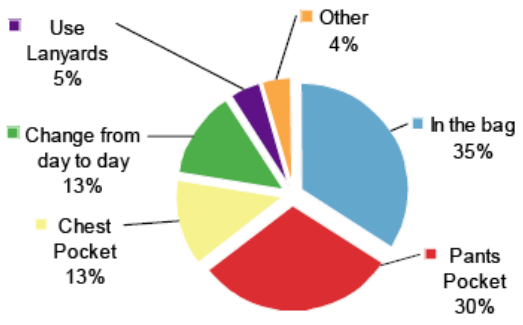


Figure 4.6. The position of smartphones in daily life [4].

In phone position classification, we focus on 4 different positions: pants' pocket, side bag, backpack, jacket/coat bag. These positions are selected according to their popularity among the users and their use in the related studies [18–20]. In [4], the results of a survey on phone usage was shared and it was mentioned that 77.6% of respondents always put their mobile phones in the bag, pants pocket, and 13.2% of people change the mobile phone position from day to day.

5. PERFORMANCE EVALUATION

In this chapter, we present the results obtained during the performance evaluation of the activity and position recognition methodology explained in Chapter 4. First, we will explain the data collection experiment setup and the system parameters, and next, we will show the results and improvements in each section.

5.1. Android Application

In order to collect activity data from the participants, a data logging application was developed in Android environment since it grants more control on the embedded sensors. It is a user-friendly application and it does not require any expertise to use. Before performing the activity, user selects the ground truth label from the list, and then puts the phone into the predetermined position (backpack, pocket, etc.) and performs the activity. Sampling rate of tge sensor manager in Android OS is selected as 100 Hz since we do not want to lose any meaningful movement pattern during data collection phase. However, this rate can be decreased in a practical application to conserve battery.

5.2. Data Collection Scenarios

The data collection phase consisted of three iterative batches. In the first stage, recognition accuracy of the system was measured by different dependency conditions, such as orientation, device and user basis, using only the accelerometer sensor. In the second stage, acceleration readings are isolated from the gravitational force and the linear acceleration data was collected by enabling the use of the gyroscope and the magnetic field sensors. This fused sensor data also made it possible to convert readings from phone coordinate system to the earth coordinate system. Hence, the same data collection scenarios were performed again with the same set of subjects to collect linear acceleration readings both in the phone and earth coordinate systems together. In the initial batch, pockets of the trousers are selected as the most suitable place to carry the

phone, as in most of the previous work [14]. In the third and final batch, 4 identical smartphones were placed on different positions, namely the trousers pocket, jacket pocket, side bag and backpack simultaneously before starting the collection of activity data. Hence, the effect of place where the phone is carried on the recognition accuracy can be evaluated with these data.

In the first two data collection sets (raw and linear acceleration data), tests were performed to measure the orientation, device and user dependency of the activity recognition system. Hence, we had 3 different data collection sessions performed by each individual:

- Orientation tests: carry the same phone in vertical/horizontal orientations to investigate orientation dependency.
- Device tests: carry the same model phones (Device A/B) in the same pocket to find the effect of the device difference, i.e. calibration, on recognition accuracy.
- Device-model tests: carry different model phones (Model X/Y) in the same pocket to measure the effect of the phone model.

For testing the user dependency, participants did not perform extra data collection but in the mentioned tests above, all participants carried the same set of devices to evaluate the user dependency and on 4 different positions to investigate the position dependency. In the final tests, 4 different phones were placed on the mentioned positions with random orientations to make experiments closer to the real-life usage scenario. In the following subsections, we explain the details for each of the data collection batches.

5.2.1. Raw Acceleration Data Collection

In the first data collection stage, 4 female and 16 male, total of 20 healthy participants between the ages 18 and 59 were asked to perform five locomotion activities. All participants were wearing trousers and phones were placed into their pockets before performing each activity. All activities were performed for 3 minutes and a total 15 minutes of raw acceleration data was collected from every participant in each step of

the experiments. Two Samsung Galaxy W and one Samsung Galaxy S3 model smart phones were used. In the orientation tests, first, participants were asked to perform the given activities while the phone was placed vertically in their pockets whereas in the second step phone was placed horizontally. In the device dependency experiments, first, subjects were asked to carry the same model devices (Two Samsung Galaxy W device) in the same pocket and then different models (One Samsung Galaxy and Galaxy S3) while they were performing the related activities again. Finally in the user dependency experiments, the data was collected from different participants by carrying the same device in the selected orientation.

5.2.2. Linear Acceleration Data Collection

In the second data collection stage, same set of participants were asked to perform the predefined set of activities with the previous conditions. After analyzing the data (results will be discussed in Section 5.3.4), since the effect of phone/device model on recognition rates in the first step of experiments are found to be negligible, these test are removed from the schedule in the linear acceleration data collection. First, a Samsung Galaxy S3 model phone is placed in vertical orientation to the side pocket of participants' trousers, then activities are re-performed while phone is horizontally placed. This time, data collection application is modified for collecting fused sensor data from accelerometer, gyroscope and magnetic field sensor. Fused sensor data are sampled according to two different coordinate systems; phone's own coordinate system and earth coordinate system using the transformation method explained in Chapter 4.

5.2.3. Data Collection with Different Phone Positions

In the final step, 3 female and 7 male, total of 10 healthy participants between ages of 22 and 51 were asked to perform the same set of activities with the previous conditions. This time, 4 identical smartphones (Samsung Galaxy S3) are used for collecting movement data from different parts of the body. In a real life scenario, users generally carry their phones in their trousers, jackets or coats. If they use bags, they usually choose side-bags or back packs. In the experiments, phones were placed in

these four locations before they start performing activities.

5.2.4. Activity Recognition Phase

In the classification phase, collected data from all individuals was evaluated separately and classification results were calculated on a subject basis. In the orientation dependency tests, while the data collected in one orientation was given as the training set, the data coming in the other orientation was used as the test data in the classifier. Again in the device dependency tests, the data collected in one device was used as the test set and the data collected with the other device as the training set. In the user dependency tests, the leave-one-out approach was used. In the position dependency tests, classifying process was performed in a user, activity and context basis. The collected data was processed offline: first, two feature sets (one is subset of another) are extracted from the raw data, then dimensionality reduction operation is performed and finally the classification was applied using the KNN and DT classifier available in the MATLAB statistics toolbox. Non-overlapping and 50% overlapping windowing approaches are applied during segmentation.

5.3. Experimental Analysis Results

In this section, we present the results of the activity recognition experiments on the collected data, following the same sequence explained in Chapter 4, first the results with raw acceleration, then the results with linear acceleration, and finally the results of the phone-position data collection experiments.

5.3.1. Analysis of Device, Device-Model, Orientation, and User Dependency with Raw Acceleration Data

In the first attempt, the recognition accuracy of the system was measured using the simple time domain features, including the mean, the variance and the standard deviation. Features of each window were calculated using the magnitude of the accelerometer axes and also each of the axes separately. As a result, a feature vector

with a length of 12 was obtained. In these experiments, the experimental parameter set included the window size and approach in the segmentation phase, feature list in the feature extraction phase and the classifiers (DT and KNN) in the classification phase. In order to see the effect of windowing approach, first data was segmented into 1 second of non-overlapping continuous parts and secondly it is divided into 50% overlapping pieces. The features calculated from each segment are sent to the dimensionality reduction step. The wide set of features extracted in the previous step was reduced to 3 features that are the standard deviation of the magnitude and z axis, also the mean of z axis. This reduction was performed by selecting the features which improve the recognition rates most using the sequential-forward selection according to KNN and DT classification algorithms separately. The results of the dimensionality reduction phase is given in Appendix A.

Table 5.1. Comparison of orientation, model, device user dependency using overlapping and not overlapping windows and the classifiers.

	Orientation		**Same Model**		**Different Model**		**User**	
Overlapping Ratio	0%	50%	0%	50%	0%	50%	0%	50%
KNN	85.8%	86.3%	97.3%	97.5%	95.7%	95.2%	90.3%	90.4%
DT	84.1%	85%	96.2%	96.3%	92.9%	93.5%	88.5%	88.9%

Table 5.1 shows the results in terms of accuracy for the different data collection scenarios, for 2 different windowing approaches and 2 different classifiers in terms of orientation-dependency, model-dependency, device dependency and user dependency using the mentioned feature list. In Table 5.1, we see that when the same model phones are carried in the same pocket, around 96-97% accuracy can be achieved while it is approximately 95% if the phone model is different. Also in the user dependency tests, 88-90% accuracy is achieved. However, the accuracy in the orientation test is lower compared to the other tests. Investigating the windowing approach, as it is seen on Table 5.1, with the 50% overlapping windows, small improvements on recognition rates can be obtained for each case. Although it is small, since it has a noticeable effect on the classification accuracy and usually the related studies also utilize over-

Table 5.2. The effect of using different window sizes on recognition rates with the simple feature set.

	Orientation			**Same Model**			**Different Model**			**User**		
Window (sec)	1.28	2.56	5.12	1.28	2.56	5.12	1.28	2.56	5.12	1.28	2.56	5.12
KNN	85.7	86.4	86.1	97.5	97.7	97.8	96	96.3	96.1	90.1	90	90.7
DT	85.6	85.8	86.1	96.3	96.5	96.1	92.6	92.5	92.7	88.8	88.6	88.9

lapping windows [14], it is decided to use 50% overlapping approach in the upcoming experiments.

We also investigated the window size on the recognition accuracy. As mentioned, initially we used 1 second window size. To investigate this issue, the data is processed with three different window sizes: 1.28 seconds, 2.56 seconds and 5.12 seconds. The reason to select such window sizes is that, when we compute the FFT-based features (using different features will be discussed next in this section), it is required to select window lengths as a power of 2. In order to do a fair comparison with using frequency domain features, whose results are presented further in this subsection, we utilize the same window sizes for the simple features. Results are presented in Table 5.2. Results are very similar to the results achieved with 1 second windows given in Table 5.1 and using different window sizes with simple feature set did not impact the results.

In the next step, we investigate whether using an extended feature set can help to improve the classification accuracy. Additionally, in order to investigate the effect of the window size, collected data were segmented into different size windows such as 1.28, 2.56 and 5.12 seconds. Since the extended feature set consists of frequency domain features, it is required to select window lengths as power of 2, in order to make these FFT based calculations. As it is mentioned for the simple time-domain features, similar steps for feature extraction and selection operations are applied to the extended feature set. As a result, frequency domain features, including the first 10 FFT coefficients, zero crossing rate, maximum correlation value and period are added

Table 5.3. The effect of using an extended feature set and window sizes on recognition rate.

	Orientation			**Same Model**			**Different Model**			**User**		
Window (sec)	1.28	2.56	5.12	1.28	2.56	5.12	1.28	2.56	5.12	1.28	2.56	5.12
KNN	87.3	87.5	85.8	97.7	97.8	97.8	96.5	96.9	97	90.5	91.1	91.3
DT	83.3	82.1	82.1	96.5	96.8	96	92.5	92.6	92.6	91.7	93	94.1

to the simple feature set to obtain the new feature set. Table 5.3 shows the recognition rates of KNN and DT classification algorithms for different 50% overlapping window sizes on each experiment.

In Table 5.3, we observe that considering the model and device dependency, results are very similar with using simple features (Table 5.2). When we compare the results for the orientation experiments, with KNN classifier, results improve by 1-2%, whereas it is similar for the DT classifier. Considering the results of user dependency, using the extended feature set with the DT classifier, increase in results vary from 2 to 5%. As these results suggest, we can achieve acceptable accuracy results with different devices and models. However, when the orientation of the phone changes, the accuracies are lower and this holds for the user tests as well.

Table 5.4. Recognition rates after applying normalization on the feature set.

	Orientation			**Same Model**			**Different Model**			**User**		
Window (sec)	1.28	2.56	5.12	1.28	2.56	5.12	1.28	2.56	5.12	1.28	2.56	5.12
KNN	90.2	91.5	92.3	98.7	98.9	99.1	98.2	98.4	98.8	96.0	96.2	96.4
DT	84.6	84.2	84.9	96.7	96.9	96.9	93.5	94.1	93.1	96.1	95.9	96.6

With the lower results obtained in orientation and user dependency tests, we focused on enhancing our classification quality. First, we investigated the basis of our classification algorithms. In our KNN implementation, we used Euclidean distance as distance metric and Euclidean distance is simply based on the straight line distance

between two points in Euclidean space. In our feature set, there are several features on different scales which may affect the classification accuracy directly in Euclidean based distance calculation. So we decided to apply normalization operation on feature set. Table 5.4 confirms our assumption that there are significant increase (5-6% in some window sizes) in recognition rates of orientation and user dependency test using KNN classifier. There is an important drawback of applying normalization on the feature set during online activity classification. Normalization operation needs huge amount of computation power which makes it less preferable choice on the smartphone based activity recognition environment.

Table 5.5. Recognition rates after applying filtering during the classification.

	Orientation			**Same Model**			**Different Model**			**User**		
Window (sec)	1.28	2.56	5.12	1.28	2.56	5.12	1.28	2.56	5.12	1.28	2.56	5.12
KNN	87.7	88.4	89.1	97.2	99.1	98.9	97.9	97.7	98.2	91.2	92.2	94.3
DT	87.1	87.2	89.2	96.6	97.7	98.2	95.2	93.6	97.4	92.2	94.0	95.2

So, we decided to move forward with another approach which has to be lightweight for the use in mobile environment. We proposed using filtering during the classification phase. Our filtering approach is simply based on labelling the class of a window by utilizing previous sequential class labels. For instance, in a set of consecutive windows, the class label of first 5 window is determined as sitting and 6th window as running. Filtering simply evaluates the class labels of these 5 windows and alters the class label 6th window as sitting. This operation is performed by weighting classification output and history information then deciding the class label according to majority voting. Table 5.5 shows that there are noticeable increases up to 4% percent in some window sizes using both classifiers.

As indicated in Table 5.3, orientation and model tests have better accuracy for 2.56 second length windows while this is true for 5.12 second windows for different model and user dependency tests. Also, even with the use of extended feature set, the success of orientation and user dependency cases are lower compared to the results

of the model tests. When we analyze the confusion matrices of orientation and user dependency tests for KNN and DT classifiers, biking activity has considerably low accuracy compared to the other activities. Confusion matrices for both classifiers for the orientation tests with 1 second windows is given in Table 5.6. Classifiers tend to confuse the walking and running activities, since performing speed of these periodic activities may vary among subjects. Biking activity is confused with walking and sitting, since the users sometimes stop while pedalling during the biking activity.

Table 5.6. Confusion matrices for orientation tests with KNN and DT classifiers.

	KNN					**DT**				
	Run	Stand	Bike	Sit	Walk	Run	Stand	Bike	Sit	Walk
Run	**83.2%**	0.3%	5.1%	0.1%	11.3%	**84.1%**	3.5%	3.4%	0.1%	8.9%
Stand	0%	**97.7%**	0%	2.3%	0%	0.2%	**97.4%**	0.1%	2.3%	0%
Bike	2.9%	4.7%	**68.9%**	7.3%	16.2%	9.4%	5.4%	**64.9%**	10.4%	10.1%
Sit	0%	4.6%	2.3%	**93.1%**	0%	2.3%	6.8%	2.4%	**88.5%**	0%
Walk	0.6%	0.6%	5.2%	0%	**93.6%**	2.0%	4.7%	4.8%	0%	**88.5%**

Table 5.7. Confusion matrices for user tests with KNN and DT classifiers.

	KNN					**DT**				
	Run	Stand	Bike	Sit	Walk	Run	Stand	Bike	Sit	Walk
Run	**84.1%**	0.3%	1.8%	0.2%	13.6%	**87.2%**	0.2%	0.7%	0.2%	11.7%
Stand	0%	**96.1%**	0%	3.9%	0%	0.2%	**95.3%**	0%	4.3%	0.2%
Bike	2.4%	0%	**94%**	1.1%	2.5%	2.2%	0%	**94.5%**	1%	2.3%
Sit	0%	4.6%	0.1%	**95.3%**	0%	0%	4.5%	0.3%	**95.2%**	0%
Walk	12.1%	0.2%	0.1%	0%	**87.6%**	11.7%	0.1%	0.7%	0%	**87.5%**

5.3.2. Analysis of Orientation, and User Dependency with Linear Acceleration Data

After expanding the feature set with the frequency domain features, as mentioned we still encountered results with lower accuracy in the orientation and user tests. The orientation of a phone can be recognized by fusing sensor data from the accelerometer, the gyroscope and the magnetic field sensor, hence, the gravity component on each axis

can be isolated to obtain the linear acceleration values. In order to measure the effect of the linear acceleration on the recognition rates, new tests are performed with the same set of 20 participants again. In these tests, the linear acceleration data referenced by the phone and earth coordinate systems were collected at the same time. Only the extended feature set is used in the classification since it provided better results in the initial tests, explained in Section 5.3.1.

As illustrated in Table 5.8, in the orientation tests the recognition accuracy increased to 91-92% with the new dataset using the linear acceleration values when the phone coordinate system was selected as the reference. Also in the user-dependency tests, the recognition accuracy increased to 93-94%. But with the use of earth coordinates, orientation dependent recognition accuracy increases to 94-95% and it inclined up to 98% for the user dependency case.

Table 5.8. Accuracy of KNN and DT classifiers for different window sizes on phone and earth coordinate systems using linear acceleration.

	Phone Coordinate System						**Earth Coordinate System**					
	Orientation			**User**			**Orientation**			**User**		
Window (sec)	1.28	2.56	5.12	1.28	2.56	5.12	1.28	2.56	5.12	1.28	2.56	5.12
KNN	90.7	91.6	91.8	92.83	92.78	94	94.65	95.05	95.2	97.25	97.7	98.6
DT	89	90.5	90.48	93.48	94.6	93.75	92.95	92.85	95.7	96.2	96.8	96.7

Confusion matrix for the KNN classifier for the orientation tests using linear acceleration is given in Table 5.9. The table shows that, when we consider using phone coordinates, the recognition rate of biking activity increased significantly compared to the results in Table 5.6. Also the recognition accuracy for running and sitting activities increased. However, there is a decrease in the recognition of standing and walking activities using phone coordinates. Similar results are observed for the DT classifier.

Before the design decision of fusing the accelerometer, the gyroscope and the

Table 5.9. Confusion matrices for orientation tests with KNN classifier using linear acceleration and different coordinate systems.

	Phone Coordinate System					**Earth Coordinate System**				
	Run	Stand	Bike	Sit	Walk	Run	Stand	Bike	Sit	Walk
Run	**99.9%**	0%	0%	0%	0.1%	**100%**	0%	0%	0%	0%
Stand	0%	**93.0%**	0%	7.0%	0%	0%	**93.2%**	0%	6.8%	0%
Bike	1.0%	0%	**89.5%**	0.8%	8.7%	0%	0%	**89.8%**	0%	10.2%
Sit	0%	5.2%	0%	**94.8%**	0%	0%	0.1%	0%	**99.9%**	0%
Walk	0%	0%	12.7%	0%	**87.3%**	0%	0%	8.6%	0%	**91.4%**

Table 5.10. Confusion matrices for user tests with KNN classifier using linear acceleration and different coordinate systems.

	Phone Coordinate System					**Earth Coordinate System**				
	Run	Stand	Bike	Sit	Walk	Run	Stand	Bike	Sit	Walk
Run	**96.2%**	0%	1.9%	0%	1.2%	**95.7%**	0%	4.3%	0%	0%
Stand	0%	**96.2%**	0%	3.8%	0%	0%	**100%**	0%	0%	0%
Bike	1.5%	0%	**96.1%**	0%	2.4%	0.5%	0%	**98.6%**	0%	0.9%
Sit	0%	8.2%	0%	**91.8%**	0%	0%	0%	0%	**100.0%**	0%
Walk	1.0%	0%	5.7%	0%	**93.3%**	0%	0%	4.8%	0%	**95.2%**

magnetic field sensors, we expected that if the movement of the smart phone could be measured according to a single reference system, the orientation in the pocket would lose its meaning. So we converted the linear acceleration readings from the phone coordinate system to the earth coordinate system. Tables 5.8, 5.9 and 5.10 confirm our assumption that results of the orientation dependency tests boosted up to 97%.

Table 5.11 summarizes the findings for the two classifiers for the 5.12 seconds window sizes in terms of accuracy and F-measure.

Table 5.11. Accuracy and F-Measure values for User and Orientation Tests.

	Phone Coordinate System				**Earth Coordinate System**			
	Orientation		**User**		**Orientation**		**User**	
Classifier	KNN	DT	KNN	DT	KNN	DT	KNN	DT
Accuracy	91.8	90.6	94.0	94.4	95.2	95.7	98.6	96.7
F-measure	92.9	87.1	94.9	94.5	94.9	93.8	98.0	96.5

5.3.3. Analysis of Position Dependency on Activity Recognition

In Section 5.3.2, we observed that removing the effect of gravity on the acceleration readings of the fused sensor output and using the earth coordinate system as a global reference point helped us to recognize activities independent from the orientation of the phone. Next, we investigate whether it is possible to predict the activities independently from the phone position/placement. In order to analyze this, we performed another batch of experiments. In the final tests, we placed 4 same-model phones on different positions, including, pocket of the trousers, jackets/coats, backpack and sidebag. Then the same set of activities are performed by the participants. We collected fused acceleration data as referencing both the coordinate systems of the phone and the earth. In the activity classification phase, we used the extended feature set.

We trained and tested the collected data according to five possible methods as follows:

- **Case 1- Person specific, generalized classifier:**
  Test: The data collected on a single location from a single user (e.g. backpack of subject 5)
  Training: The data collected on other locations from the same user (e.g. trousers, jacket and sidebag of subject 5) excluding the test position
- **Case 2. Person and position specific classifier:** Cross validation in between same position from all users.

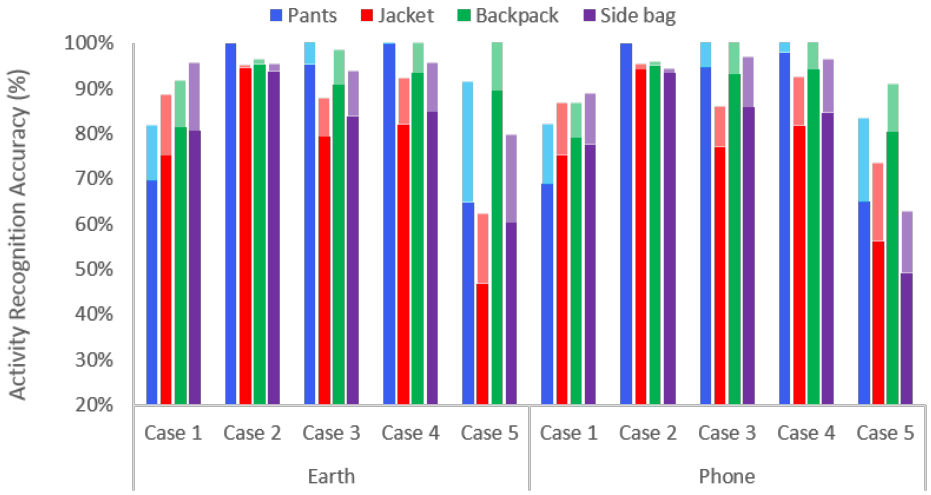


Figure 5.1. Activity recognition accuracy of position dependency tests using phone and earth coordinate systems for KNN.

- **Case 3 – Generalized classifier (user independent)**

  Test: The data collected on a single location from a single user (e.g. jacket of subject 4)

  Training: The data collected on all locations from the other users (e.g. trousers, jacket, sidebag and backpack of other subjects)
- **Case 4 – Position specific classifier (user independent)**

  Test: The data collected on a single location from a single user (e.g. sidebag of subject 7)

  Training: The data collected on same location from the other users (e.g. sidebag of other subjects)
- **Case 5 – Joint classifier (user independent):** Each activity and position pair is associated with a class

  Test: The data collected from a single user in all locations and activity-position pairs are created

  Training:The data collected from other participants are used to create same classes.

Figures 5.1 and 5.2 show the results in terms of accuracy for 5 different cases, both when phone coordinates are used and when earth coordinates are used. Both in

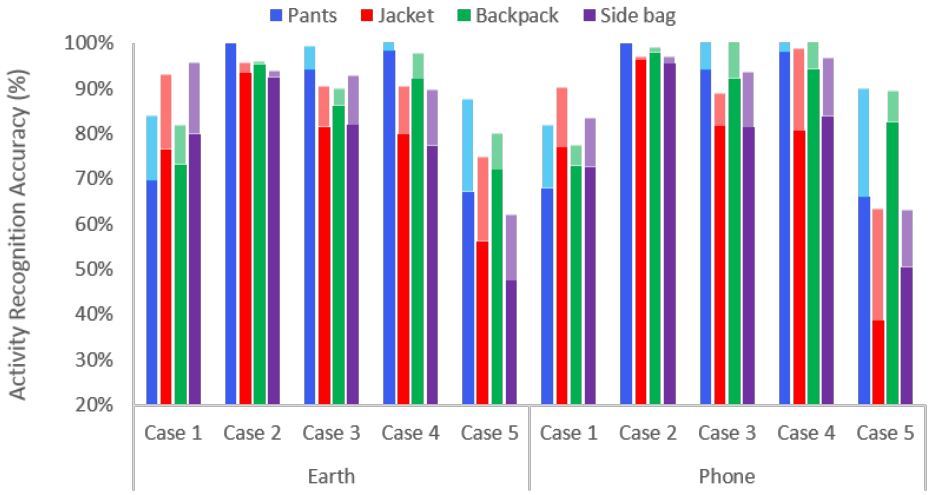


Figure 5.2. Activity recognition accuracy of position dependency tests using phone and earth coordinate systems for DT.

Figure 5.1 and Figure 5.2, when we compare the results of Case 1 and Case 2 (person specific training), it is clear that using position specific classifiers exhibit better performance compared to a generalized classifier. On the other hand, it is costly to collect training data from all users for each position and person-independent classification is often necessary in practical applications. Comparing the performance results in Cases 3 and 4 (person independent), they are very similar. Looking at the KNN results, with using phone coordinates, on average 94.7% for pants, 77.1% for jacket, 93.9% for backpack and 85% for side bag positions are achieved for Case 3 (generalized classifier). Similarly, on average 98% for pants, 81.8% for jacket, 94.2% for backpack and 84.7% for side bag positions are achieved for Case 4 (position-specific classifier). Results are also very similar when earth coordinates are considered. The highest difference between using a position-specific classifier and a generalized classifier is for the pants position: 3.3% difference for phone coordinates and 4.5% for the earth coordinates. For the jacket position, using position-specific model increases the accuracy by 4.5% again using the phone coordinates and 2.7% when using earth coordinates. For the backpack position the difference is 1% and 2.6% for phone an earth coordinates respectively. For the sidebag position, when phone coordinates are used the generalized classifier performed better by 1%, while the position-specific classifier performed 1% better with earth coordinates. Also when position specific training data is supplied, position pants

has the highest recognition accuracy compared to other positions. On average, with KNN, considering all the 4 positions 2.7% increase in accuracy is achieved with using a position-specific classification compared to using a generalized classifier with earth coordinates.

If we look at the accuracy results of the DT classifier in Figure 5.2, similar trends are observed for pants and backpack positions with the results of the KNN classifier. However, for the jacket position using a generalized classifier performed better by 1% using both the coordinate systems. In the side-bag position, using position-specific classifier performed better by 2%, while the generalized classifier performed 4.5% better using the earth coordinate system On average, with DT, considering all the 4 positions 0.95% increase in accuracy is achieved with using a position-specific classification compared to using a generalized classifier with earth coordinates.

These findings are consistent with previous results found in the literature [16, 17] while it contradicts with the results of [18, 19] where it was reported that position-specific classification significantly improves the results. Although in some cases using a position-specific classifier performs better, one should consider that in order to perform position-specific classification in a practical application, one should first identify the phone position and then use the model associated with that position for activity classification. Phone position identification is also a challenging problem as we investigate in Section 5.3.3

Joint classification was reported to achieve higher accuracies compared to a generalized classifier and position-specific classifier in [20]. However, the activity set was smaller and the transitions between the activities was targeted in that study with different classifiers. Here, using a joint classifier did not contribute to increasing accuracies. With the KNN classifier, 65% accuracy for pants, 56% for jacket, 80% for backpack and 50% for the sidebag positions are achieved when phone coordinates are used. Similarly, 65% accuracy for pants, 47% for jacket, 89% for backpack and 60% for the sidebag positions are achieved when earth coordinates are used. These results are much lower than the results achieved with a generalized classifier and a position-specific

classifier. Similar trends are observed for the DT classifier as well.

Table 5.12. Confusion matrices of jacket position using phone and earth coordinates.

Jacket										
	Phone Coordinates					**Earth Coordinates**				
	Run	Stand	Bike	Sit	Walk	Run	Stand	Bike	Sit	Walk
Run	**98.2**%	0%	0%	0%	1.8%	**98.8%**	0%	0%	0%	1.2%
Stand	0%	**68.5%**	0%	31.5%	0%	0%	**58.3%**	0%	41.7%	0%
Bike	0%	0%	**96.7%**	0%	3.3%	0%	0%	**100.0%**	0%	0%
Sit	0%	54.5%	0%	**45.5%**	0%	0%	46.3%	0%	**53.7%**	0%
Walk	0%	0.0%	0%	0%	**100.0%**	0%	0%	0%	0%	**100.0 %**

Although the recognition accuracy for the pants positions is very high in both Cases 3 and 4, it is relatively lower for the other three positions. In order to elaborate on this issue, we also looked at the confusion matrices for Case 4 (position-specific classification) for the jacket, backpack and sidebag positions. The confusion matrices are presented in Tables 5.12, 5.13, 5.14. We do not present confusion matrix of the pants position because of its high recognition accuracy. When we investigate the low accuracy in these 3 positions, confusion matrices indicate that KNN classifier is good at recognizing dynamic activities but it is far more worst for static activities. As expected, sitting and standing activities are highly misclassified because of their static movement characteristics. It is hard to infer whether a person is sitting or standing when the phone is located in a bag or in a jacket pocket.

With the low results originated from static activities, we decided to merge sitting and standing activities in the next round naming it as “stationary”. This time the number of activities decreased from 5 to 4 and these are walking, running, biking and stationary (sitting/standing). Figure 5.3 and Figure 5.4 show that with the merge of sitting and standing activities, recognition accuracy increased to 98% in all positions for Case 3 and Case 4. In this case again, for the pants position when a position-specific classifier is used the accuracy is 2% higher than using a generalized KNN classifier. However, this is not true for the other positions and even for the side-bag position, using a generalized classifier is 0.8% better. In the DT case, using a generalized classifier performed at most 1.5% better than a generalized classifier. Hence, if we take into

Table 5.13. Confusion matrices of backpack position using phone and earth coordinates.

Backpack										
	Phone Coordinates					**Earth Coordinates**				
	Run	Stand	Bike	Sit	Walk	Run	Stand	Bike	Sit	Walk
Run	**100.0%**	0%	0%	0%	0%	**100.0%**	0%	0%	0%	0%
Stand	0%	**97.6%**	0%	2.4%	0%	0%	**85.9%**	0%	14.1%	0%
Bike	0%	0%	**100.0%**	0%	0%	0%	0%	**100.0%**	0%	0%
Sit	0%	22.6%	0%	**77.4%**	0%	0%	17.5%	0%	**82.5%**	0%
Walk	0%	0%	0%	0%	**100.0%**	0%	0%	0%	0%	**100.0%**

Table 5.14. Confusion matrices of sidebag position using phone and earth coordinates.

Side Bag										
	Phone Coordinates					**Earth Coordinates**				
	Run	Stand	Bike	Sit	Walk	Run	Stand	Bike	Sit	Walk
Run	**99.8%**	0%	0%	0%	0.2%	**99.6%**	0%	0%	0%	0.4%
Stand	0%	**80.3%**	0%	19.7%	0%	0%	**67.9%**	0%	32.1%	0%
Bike	0%	0%	**95.1%**	0%	4.9%	0%	0%	**89.9%**	0%	10.1%
Sit	0%	46.7%	0%	**53.3%**	0%	0%	30.3%	0%	**68.7%**	0%
Walk	0%	0%	2.1%	0%	**97.9%**	0%	0%	3.1%	0%	**96.7%**

account all the results, it is difficult to say whether a position-specific classifier performs better than a generalized classifier.

For Case 5, joint classifier, there is a moderate increase around 8% in recognition accuracy using both coordinate systems with KNN and DT classifiers. However, still the results with a joint classifier is much below the results with using a generalized or position-specific classifier. So in the next step, for the joint classifier (Case 5), we decided to change our evaluation methodology. Sitting while the phone in backpack or sitting while the phone in side bag show very similar characteristics in terms of features for the classifier. Since our primary objective is accurate classification of the activities, we decided to re-evaluate the output of our classifiers. This time when an activity

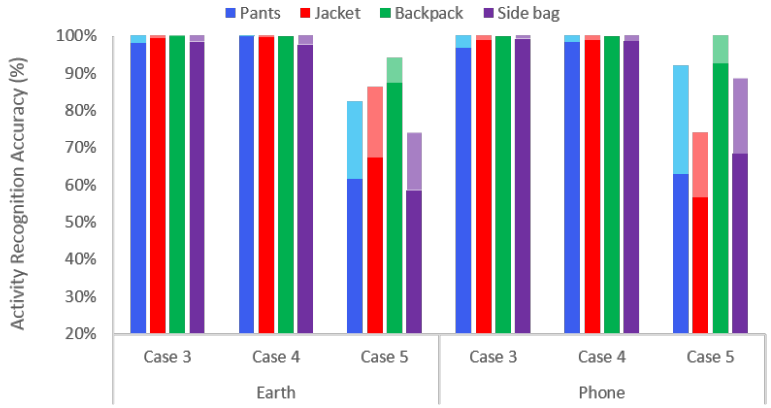


Figure 5.3. Activity recognition accuracy of position dependency tests using phone and earth coordinate systems with KNN for 4 activities.

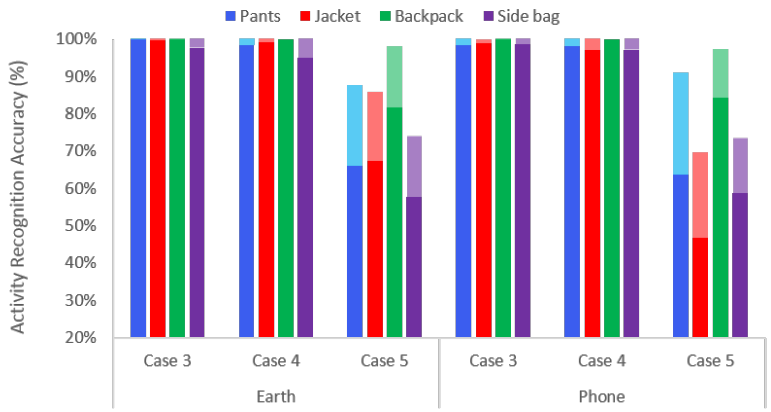


Figure 5.4. Activity recognition accuracy of position dependency tests using phone and earth coordinate systems with DT for 4 activities.

is classified accurately, it will be evaluated as true whether its position is classified wrong. If walking activity is recognized accurately, it will be evaluated as true no matter which position it is carried. This time we defined 5 activities set as Case 5.1 and 4 activities set as Case 5.2 where sitting and standing activities are merged as standalone stationary class. Figure 5.5 shows the overall recognition accuracy achieved with the new activity set. For Case 5.1, recognition rates increased above 80% for both classifiers. With the merge of sitting and standing activity in Case 5.2, recognition accuracy increased above 98% for KNN and 96% for DT classifiers.

Figures 5.6 and 5.7 show the results in terms of accuracy for subcases 5.1 and 5.2 on position basis. In Case 5.1 with the KNN classifier, 94% accuracy for pants, 80% for

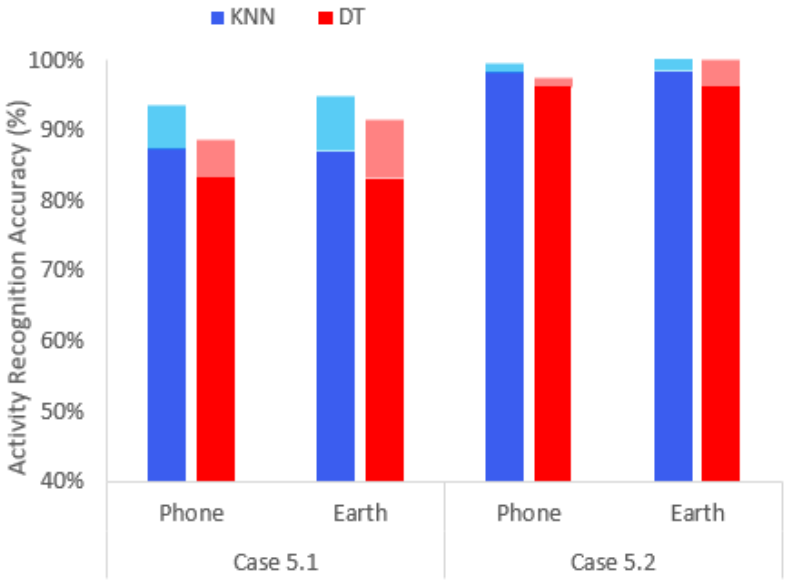


Figure 5.5. Activity recognition accuracy of joint classification (Case 5).

jacket, 91% for backpack and 83% for the sidebag positions are achieved when earth coordinates are used. Similarly, 94% accuracy for pants and backpack, 77% for jacket, 86% for the sidebag positions are achieved when phone coordinates are used. DT classifier performs slightly lower performance compared to KNN. When the number activities decreased from 5 to 4, the recognition accuracy increased significantly for both coordinates. With the KNN classifier, 97% accuracy for pants, 99% for jacket and backpack and 98% for the sidebag positions are achieved when earth coordinates are used. Similar results obtained with the phone coordinate system.

5.3.4. Analysis of Position Recognition

As discussed in Section 5.3.3, position-specific classification does not always improve the accuracy compared to using a generalized classifier. Moreover, for position specific classification, in a practical application, first the position of the phone should be detected and then the specific classifier should be run. In this section, we investigate the phone position recognition using the same set of extended features and the KNN and DT classifiers.

While Figure 5.8 shows the average position recognition accuracy for both of the

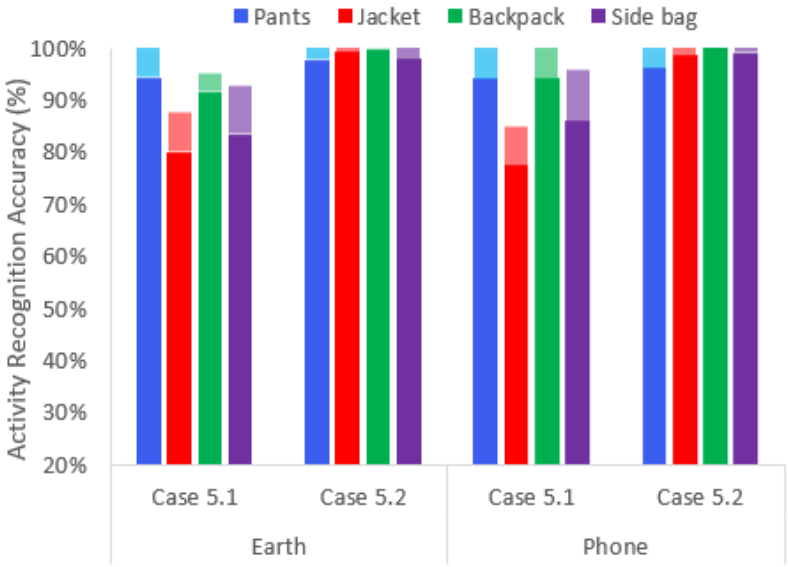


Figure 5.6. Activity recognition accuracy of joint classification with KNN (Case 5).

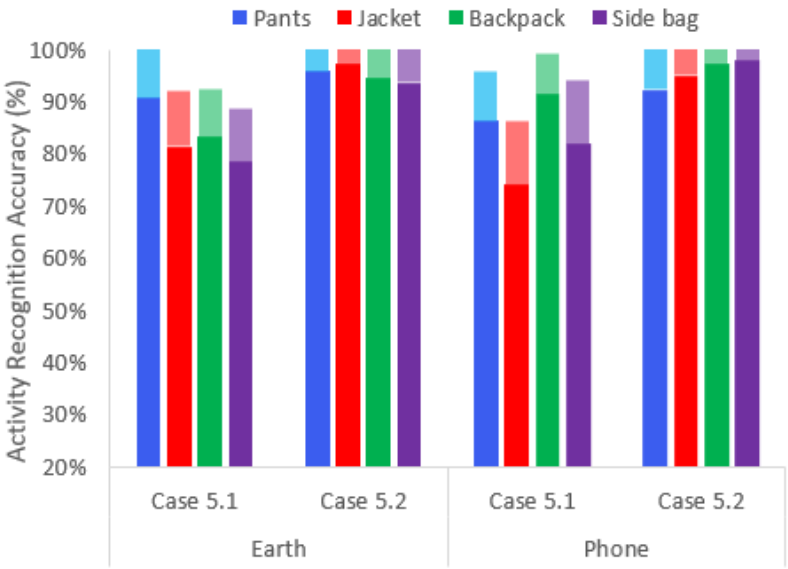


Figure 5.7. Activity recognition accuracy of joint classification with DT (Case 5).

classifiers and both of the coordinate systems, Figure 5.9 shows the activity-specific accuracies. Around 71.5% accuracy is achieved for position recognition with the KNN classifier, and 69% accuracy is achieved with the DT classifier. When we investigate the activity specific results we see that positions can be identified with a higher accuracy (81.8%:KNN, 84.7%:DT) when the users are standing if the phone coordinates are used. However, when the earth coordinates are used, with walking and biking activities, phone positions are identified with a higher accuracy, 76.8%: biking-

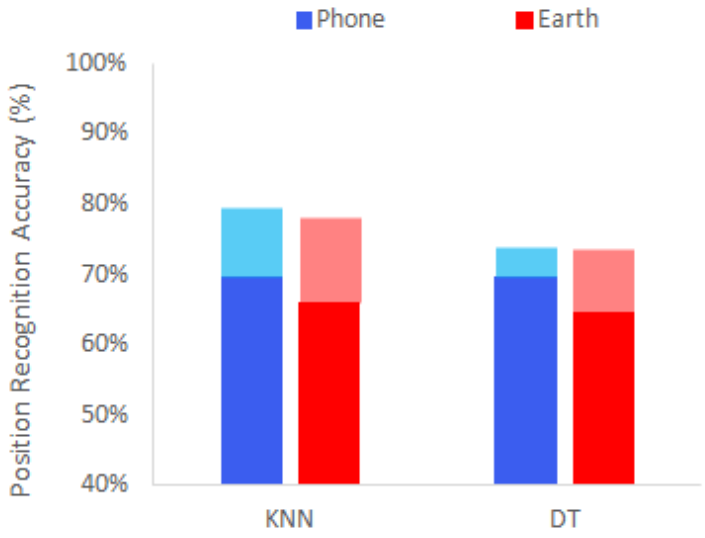


Figure 5.8. Position recognition accuracy using phone and earth coordinate system for both classifiers.

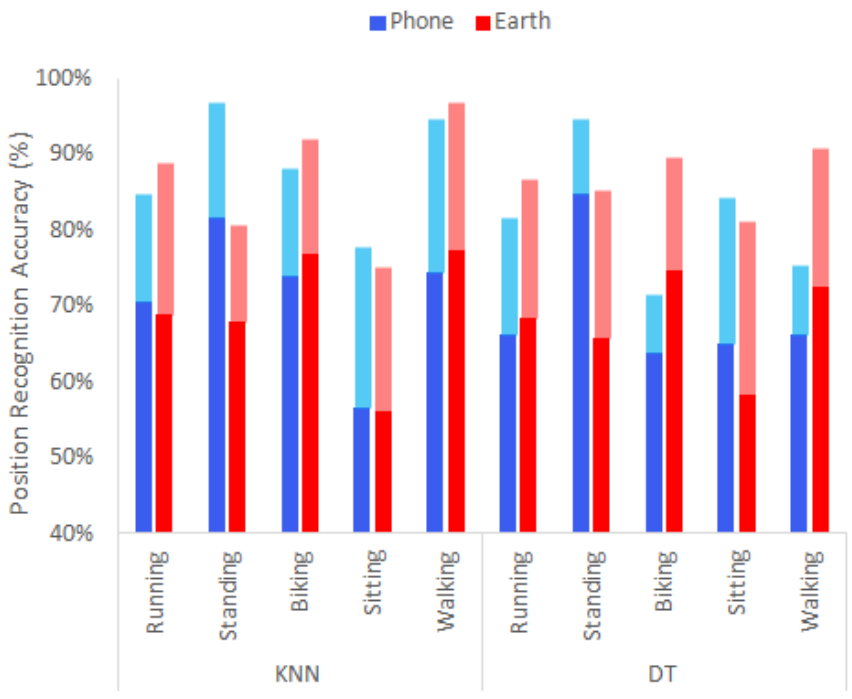


Figure 5.9. Position recognition accuracy on activity basis using KNN and DT classifiers.

KNN, 77.3%:walking-KNN, 74.6%: biking-DT, 72.4%: walking-DT. In Figure 5.10, we present the recognition accuracies for each position separately. Backpack position is recognized with a higher accuracy compared to the other positions by both of the

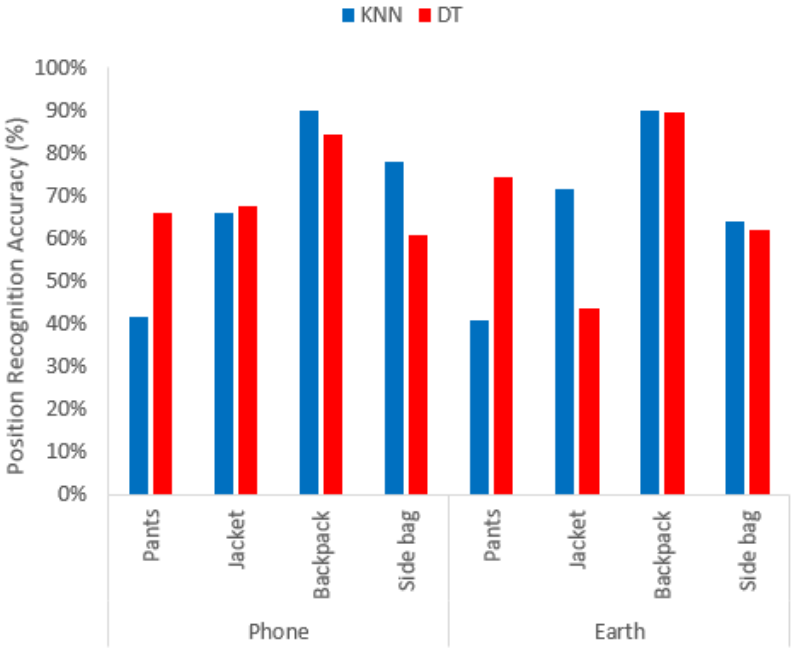


Figure 5.10. Average position recognition accuracy for each position.

classifiers whereas the accuracies vary for different positions by the two classifiers.

When we compare these results with the results achieved in the literature, in [18], light, proximity, magnetometer, gyroscope, gravity, acceleration and linear acceleration sensors were employed and 92.94% accuracy with best set of all features and 66% accuracy with a selected set of features was achieved. However, in our study we only utilize the sensors that are used in the activity recognition stage. In [21], the focus was on in-pocket versus out-of pocket detection and using only the microphone, 80% accuracy was reported to detect these two classes. The use of accelerometer for position recognition was discussed in [17], where bag, ear, hand and pocket positions were considered. The average accuracy was reported as 94%, however, the activity set included only walking.

As mentioned, although position specific activity recognition, or knowing the positions, does not significantly improve the results compared to a generalized classifier position recognition should be further investigated for other purposes such as the volume of an incoming call automatically can be increased and vibrations can be turned on if phone is detected in the pocket.

6. CONCLUSION

In this thesis, we focused on the challenges of practical activity recognition on smart phones. We specifically concentrated on the challenges arising from the differences in user behavior and device, model differences as well as the burden of the training phase when using statistical machine learning algorithms in the classification of activities. Using the accelerometer, the magnetic field sensor and the gyroscope on the phones, we explored the recognition of simple locomotion activities on a device, user, orientation and position-independent way. With the data collection experiments, performed with 20 users, we showed that although the accuracies are quite high in user and device dependency tests, the difference in the orientation of the phone decreases the accuracy. To improve this, we proposed to use the linear acceleration, excluding the effect of gravitational force using the earth coordinates with the help of the gyroscope and the magnetic sensor. Additionally, we used signal transformation and converted the sensor readings from phone coordinates to earth coordinates. With these methods, we showed that the accuracy results increased remarkably. As the last step, we analyzed the position-dependency to see whether position-specific recognition can perform better than a generalized classifier. Besides position-independent and position-specific methods, we also investigated the performance with joint classification where activities and positions are associated with a specific class. As a result of these experiments, we showed that position-specific recognition is not necessary, a generalized classifier also performs similarly. However, we observed that separating the stationary activities, such as sitting and standing, is very difficult in some positions, other than the pants pocket. Accordingly, we analyzed the performance when these activities are combined into a single class and showed the recognition accuracies significantly increase for these phone positions. Although it is not the main topic of this study, we also analyzed the phone position detection performance using a similar methodology to the activity recognition, and showed that around 70% accuracy can be achieved, and this should be further explored.

As a future work, we will investigate the use of our findings on a practical activity recognition application running on a phone all day long where users are not given a predefined scenario as in our tests. We will investigate the performance using different metrics, especially in such a practical setting, correctly estimating the duration of the activities would be very important. Additionally, we will be investigating the battery consumption performance of such a continuous sensing application.

APPENDIX A: FEATURE SELECTION DETAILS

In the each step of experiments, feature selection phase was performed again in order to select the fittest feature set. In this chapter, the selection rates of features which are used in the classification phase are presented. In order to add a feature to corresponding feature set, it must appear in at least 20% of the selected features set. In the raw acceleration experiments, feature selection algorithm is executed for orientation, device and user dependency tests separately and the average of selection rates were calculated. If a feature is selected 20% or more in average, it is added to feature set. These steps were also performed for linear acceleration and position dependency tests as well. Selected features are highlighted with bold characters in the average columns.

Table A.1. Selection rates of features with 2.56 second windows and KNN classifier for raw acceleration experiments.

			Orientation	Same Model	Different Model	User	Average
Extended Features	Simple Features	μ_m	0.0%	15.0%	15.0%	17.5%	11.9%
		σ_m^2	20.0%	12.5%	15.0%	12.5%	15.0%
		σ_m	32.5%	60.0%	37.5%	42.5%	**43.1%**
		μ_x	0.0%	10.0%	12.5%	27.5%	12.5 %
		σ_x^2	5.0%	7.5%	5.0%	17.5%	8.8%
		σ_x	20.0%	5.0%	5.0%	7.5%	9.4%
		μ_y	2.5%	7.5%	20.0%	22.5%	13.1%
		σ_y^2	7.5%	5.0%	7.5%	15.0%	8.8%
		σ_y	30.0%	7.5%	17.5%	5.0%	15.0%
		μ_z	100.0%	20.0%	22.5%	47.5%	**47.5**%
		σ_z^2	17.5%	17.5%	12.5%	15.0%	15.6%
		σ_z	55.0%	12.5%	17.5%	25.0%	**27.5%**
		zcr_m	62.5%	37.5%	30.0%	55.0%	**46.3%**
		$I_{max_{corr}}$	40.0%	22.5%	35.0%	42.5%	**35.0%**
		T_m	77.5%	50.0%	45.0%	65.0%	**59.4%**
		F_{c_1}	0.0%	50.0%	40.0%	50.0%	**35.0%**
		F_{c_2}	27.5%	47.5%	65.0%	25.0%	**41.3%**
		F_{c_3}	22.5%	65.0%	72.5%	35.0%	**48.8%**
		F_{c_4}	12.5%	50.0%	37.5%	42.5%	**35.6%**
		F_{c_5}	5.0%	32.5%	27.5%	35.0%	**25.0%**
		F_{c_6}	27.5%	45.0%	45.0%	45.0%	**40.6%**
		F_{c_7}	22.5%	20.0%	32.5%	12.0%	**21.8%**
		F_{c_8}	47.5%	42.5%	30.0%	35.0%	**38.8%**
		F_{c_9}	37.5%	37.5%	37.5%	42.5%	**38.8%**
		$F_{c_{10}}$	47.5%	30.0%	22.5%	40.0%	**35.0%**

Table A.2. Selection rates of features with 2.56 second windows and DT classifier for raw acceleration experiments.

			Orientation	Same Model	Different Model	User	Average
Extended Features	Simple Features	μ_m	0.0%	8.0%	7.0%	12.0%	6.8%
		σ_m^2	5.0%	17.5%	12.5%	0.0%	8.8%
		σ_m	42.5%	67.5%	42.5%	35.0%	**46.9%**
		μ_x	0.0%	12.5%	20.0%	30.0%	15.6%
		σ_x^2	15.0%	10.0%	20.0%	30.0%	18.8%
		σ_x	5.0%	2.5%	2.5%	0.0%	2.5%
		μ_y	5.0%	2.5%	15.0%	30.0%	13.1%
		σ_y^2	12.5%	17.5%	22.5%	17.5%	17.5%
		σ_y	5.0%	0.0%	2.5%	0.0%	1.9%
		μ_z	87.5%	15.0%	22.5%	47.5%	**43.1%**
		σ_z^2	15.0%	10.0%	17.5%	12.5%	13.8%
		σ_z	47.0%	0.0%	40.0%	36.0%	**30.8%**
		zcr_m	52.5%	50.0%	32.5%	55.0%	**47.5%**
		$I_{max_{corr}}$	37.5%	22.5%	25.0%	35.0%	**30.0%**
		T_m	85.0%	52.5%	62.5%	85.0%	**71.3%**
		F_{c_1}	2.5%	57.5%	47.5%	60.0%	**41.9%**
		F_{c_2}	32.5%	25.0%	30.0%	37.5%	**31.3%**
		F_{c_3}	30.0%	50.0%	47.5%	50.0%	**44.4%**
		F_{c_4}	30.0%	27.5%	17.5%	50.0%	**31.3%**
		F_{c_5}	15.0%	12.5%	30.0%	42.5%	**25.0%**
		F_{c_6}	17.5%	32.5%	27.5%	32.5%	**27.5%**
		F_{c_7}	25.0%	10.0%	17.5%	32.5%	**21.3%**
		F_{c_8}	32.5%	20.0%	17.5%	47.5%	**29.4%**
		F_{c_9}	32.5%	10.0%	22.5%	32.5%	**24.4%**
		$F_{c_{10}}$	35.0%	12.5%	10.0%	30.0%	**21.9%**

Table A.3. Selection rates of features with 2.56 second windows and KNN classifier for linear acceleration experiments.

			Phone Coordinates		Earth Coordinates		
			Orientation	User	Orientation	User	Average
Extended Features	Simple Features	μ_m	70.0%	30.0%	85.0%	80.0%	**66.3%**
		σ_m^2	15.0%	15.0%	0.0%	10.0%	10.0%
		σ_m	17.5%	50.0%	20.0%	25.0%	**28.1%**
		μ_x	20.0%	55.0%	35.0%	20.0%	**32.5%**
		σ_x^2	7.5%	25.0%	10.0%	15.0%	14.4%
		σ_x	20.0%	25.0%	5.0%	10.0%	15.0%
		μ_y	17.5%	52.5%	20.0%	25.0%	**28.8%**
		σ_y^2	5.0%	25.0%	10.0%	5.0%	11.3%
		σ_y	15.0%	27.5%	5.0%	0.0%	11.9%
		μ_z	40.0%	27.5%	30.0%	25.0%	**30.6%**
		σ_z^2	27.5%	17.5%	0.0%	10.0%	13.8%
		σ_z	17.5%	17.5%	45.0%	40.0%	**30.0%**
		zcr_m	20.0%	30.0%	25.0%	20.0%	**23.8%**
		$I_{max_{corr}}$	10.0%	22.5%	30.0%	25.0%	**21.9%**
		T_m	77.5%	75.0%	45.0%	80.0%	**69.4%**
		F_{c_1}	75.0%	87.5%	80.0%	90.0%	**83.1%**
		F_{c_2}	15.0%	30.0%	65.0%	25.0%	**33.8%**
		F_{c_3}	7.5%	17.5%	15.0%	5.0%	11.3%
		F_{c_4}	15.0%	15.0%	10.0%	15.0%	13.8%
		F_{c_5}	7.5%	22.5%	35.0%	5.0%	17.5%
		F_{c_6}	7.5%	10.0%	30.0%	10.0%	14.4%
		F_{c_7}	5.0%	15.0%	15.0%	5.0%	10.0%
		F_{c_8}	10.0%	15.0%	0.0%	0.0%	6.3%
		F_{c_9}	7.5%	15.0%	5.0%	5.0%	8.1%
		$F_{c_{10}}$	15.0%	12.5%	25.0%	20.0%	18.1%

Table A.4. Selection rates of features with 2.56 second windows and DT classifier for linear acceleration experiments.

			Phone Coordinates		Earth Coordinates		
			Orientation	User	Orientation	User	Average
Extended Features	Simple Features	μ_m	70.0%	67.5%	75.0%	75.0%	**71.9%**
		σ_m^2	5.0%	2.5%	15.0%	0.0%	5.6%
		σ_m	25.0%	32.5%	25.0%	15.0%	**24.4%**
		μ_x	5.0%	55.0%	0.0%	10.0%	17.5%
		σ_x^2	10.0%	57.5%	15.0%	75.0%	**39.4%**
		σ_x	0.0%	0.0%	0.0%	0.0%	0.0%
		μ_y	7.5%	45.0%	15.0%	5.0%	18.1%
		σ_y^2	12.5%	45.0%	20.0%	20.0%	**24.4%**
		σ_y	2.5%	0.0%	5.0%	0.0%	1.9%
		μ_z	32.5%	42.5%	30.0%	40.0%	**36.3%**
		σ_z^2	7.5%	2.5%	15.0%	10.0%	8.8%
		σ_z	27.5%	32.5%	15.0%	30.0%	**26.3%**
		zcr_m	20.0%	42.5%	25.0%	30.0%	**29.4%**
		$I_{max_{corr}}$	30.0%	27.5%	10.0%	15.0%	**20.6%**
		T_m	77.5%	90.0%	90.0%	95.0%	**88.1%**
		F_{c_1}	75.0%	87.5%	75.0%	90.0%	**81.9%**
		F_{c_2}	15.0%	27.5%	15.0%	30.0%	**21.9%**
		F_{c_3}	7.5%	17.5%	5.0%	5.0%	8.8%
		F_{c_4}	15.0%	25.0%	5.0%	0.0%	11.3%
		F_{c_5}	7.5%	25.0%	0.0%	5.0%	9.4%
		F_{c_6}	7.5%	20.0%	20.0%	0.0%	11.9%
		F_{c_7}	5.0%	7.5%	5.0%	5.0%	5.6%
		F_{c_8}	10.0%	15.0%	0.0%	5.0%	7.5%
		F_{c_9}	7.5%	25.0%	5.0%	0.0%	9.4%
		$F_{c_{10}}$	15.0%	15.0%	15.0%	5.0%	12.5%

Table A.5. Selection rates of features with 2.56 second windows and KNN classifier for position independent activity recognition experiments.

			Phone Coordinates				Earth Coordinates				
			Pants	Jacket	B.pack	S.bag	Pants	Jacket	B.pack	S.bag	Average
Extended Features	Simple Features	μ_m	80.0%	40.0%	70.0%	50.0%	80.0%	40.0%	30.0%	30.0%	**52.5%**
		σ_m^2	20.0%	0.0%	10.0%	30.0%	20.0%	10.0%	20.0%	30.0%	17.5%
		σ_m	50.0%	10.0%	20.0%	50.0%	50.0%	30.0%	30.0%	60.0%	**37.5%**
		μ_x	30.0%	70.0%	80.0%	40.0%	20.0%	50.0%	30.0%	40.0%	**45.0%**
		σ_x^2	0.0%	10.0%	20.0%	0.0%	10.0%	20.0%	20.0%	20.0%	12.5%
		σ_x	0.0%	60.0%	30.0%	30.0%	10.0%	40.0%	30.0%	40.0%	**30.0%**
		μ_y	10.0%	40.0%	10.0%	30.0%	10.0%	30.0%	20.0%	20.0%	**21.3%**
		σ_y^2	0.0%	10.0%	0.0%	30.0%	0.0%	0.0%	0.0%	10.0%	6.3%
		σ_y	20.0%	40.0%	0.0%	40.0%	30.0%	40.0%	10.0%	30.0%	**26.3%**
		μ_z	30.0%	30.0%	20.0%	10.0%	20.0%	40.0%	40.0%	20.0%	**26.3%**
		σ_z^2	0.0%	10.0%	0.0%	10.0%	0.0%	10.0%	0.0%	30.0%	7.5%
		σ_z	20.0%	20.0%	10.0%	50.0%	20.0%	10.0%	50.0%	60.0%	**30.0%**
		zcr_m	30.0%	30.0%	20.0%	10.0%	20.0%	50.0%	90.0%	90.0%	**42.5%**
		$I_{max_{corr}}$	10.0%	10.0%	0.0%	10.0%	30.0%	70.0%	80.0%	90.0%	**37.5%**
		T_m	10.0%	20.0%	10.0%	50.0%	40.0%	80.0%	80.0%	90.0%	**47.5%**
		F_{c_1}	100.0%	50.0%	90.0%	40.0%	100.0%	100.0%	100.0%	100.0%	**85.0%**
		F_{c_2}	50.0%	30.0%	90.0%	70.0%	40.0%	40.0%	70.0%	70.0%	**57.5%**
		F_{c_3}	40.0%	60.0%	60.0%	30.0%	30.0%	40.0%	60.0%	70.0%	**48.8%**
		F_{c_4}	30.0%	40.0%	30.0%	60.0%	20.0%	20.0%	60.0%	70.0%	**41.3%**
		F_{c_5}	30.0%	50.0%	30.0%	60.0%	30.0%	50.0%	60.0%	70.0%	**47.5%**
		F_{c_6}	40.0%	50.0%	30.0%	40.0%	30.0%	30.0%	50.0%	60.0%	**41.3%**
		F_{c_7}	20.0%	20.0%	20.0%	30.0%	10.0%	30.0%	30.0%	30.0%	**23.8%**
		F_{c_8}	30.0%	30.0%	40.0%	60.0%	20.0%	20.0%	50.0%	60.0%	**38.8%**
		F_{c_9}	10.0%	30.0%	20.0%	40.0%	0.0%	30.0%	50.0%	50.0%	**28.8%**
		$F_{c_{10}}$	0.0%	50.0%	40.0%	30.0%	0.0%	10.0%	50.0%	60.0%	**30.0%**

Table A.6. Selection rates of features with 2.56 second windows and DT classifier for position independent activity recognition experiments.

			Phone Coordinates				Earth Coordinates				
			Pants	Jacket	B.pack	S.bag	Pants	Jacket	B.pack	S.bag	Average
Extended Features	Simple Features	μ_m	60.0%	50.0%	100.0%	100.0%	70.0%	70.0%	100.0%	100.0%	**81,25%**
		σ_m^2	10.0%	0.0%	0.0%	20.0%	0.0%	60.0%	20.0%	20.0%	16,25%
		σ_m	30.0%	20.0%	40.0%	80.0%	10.0%	0.0%	50.0%	50.0%	**35,00%**
		μ_x	30.0%	70.0%	100.0%	100.0%	20.0%	50.0%	70.0%	70.0%	**63,75%**
		σ_x^2	0.0%	0.0%	10.0%	30.0%	0.0%	70.0%	30.0%	0.0%	17,50%
		σ_x	20.0%	20.0%	40.0%	60.0%	30.0%	0.0%	60.0%	100.0%	**41,25%**
		μ_y	10.0%	70.0%	80.0%	90.0%	20.0%	50.0%	50.0%	60.0%	**53,75%**
		σ_y^2	0.0%	0.0%	10.0%	20.0%	0.0%	30.0%	30.0%	30.0%	15,00%
		σ_y	20.0%	10.0%	30.0%	50.0%	20.0%	0.0%	40.0%	50.0%	**27,50%**
		μ_z	50.0%	50.0%	90.0%	90.0%	40.0%	70.0%	90.0%	90.0%	**71,25%**
		σ_z^2	0.0%	0.0%	20.0%	30.0%	0.0%	20.0%	30.0%	0.0%	12,50%
		σ_z	30.0%	20.0%	30.0%	50.0%	10.0%	0.0%	50.0%	100.0%	**36,25%**
		zcr_m	40.0%	50.0%	80.0%	100.0%	20.0%	50.0%	90.0%	100.0%	**66,25%**
		$I_{max_{corr}}$	70.0%	100.0%	100.0%	100.0%	50.0%	100.0%	100.0%	100.0%	**90,00%**
		T_m	90.0%	90.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	**97,50%**
		F_{c_1}	90.0%	90.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	**97,50%**
		F_{c_2}	0.0%	20.0%	30.0%	40.0%	0.0%	10.0%	70.0%	70.0%	**30,00%**
		F_{c_3}	0.0%	20.0%	40.0%	40.0%	0.0%	30.0%	50.0%	60.0%	**30,00%**
		F_{c_4}	10.0%	30.0%	30.0%	60.0%	0.0%	20.0%	30.0%	70.0%	**31,25%**
		F_{c_5}	10.0%	20.0%	30.0%	30.0%	0.0%	10.0%	40.0%	60.0%	**25,00%**
		F_{c_6}	0.0%	30.0%	50.0%	50.0%	0.0%	20.0%	20.0%	50.0%	**27,50%**
		F_{c_7}	0.0%	10.0%	30.0%	50.0%	0.0%	0.0%	10.0%	20.0%	15,00%
		F_{c_8}	0.0%	30.0%	40.0%	50.0%	0.0%	20.0%	20.0%	40.0%	**25,00%**
		F_{c_9}	0.0%	30.0%	40.0%	50.0%	0.0%	20.0%	30.0%	30.0%	**25,00%**
		$F_{c_{10}}$	0.0%	40.0%	70.0%	70.0%	0.0%	10.0%	20.0%	40.0%	**31,25%**

Table A.7. Selection rates of features with 2.56 second windows using KNN and DT classifiers for position recognition experiments.

			Phone Coordinates		Earth Coordinates			
			KNN	DT	KNN	DT	Average KNN	Average DT
Extended Features	Simple Features	μ_m	50.0%	20.0%	50.0%	40.0%	**50.0%**	**30.0%**
		σ_m^2	20.0%	0.0%	20.0%	10.0%	20.0%	5.0%
		σ_m	60.0%	50.0%	60.0%	40.0%	**60.0%**	**45.0%**
		μ_x	50.0%	40.0%	40.0%	55.0%	**45.0%**	**47.5%**
		σ_x^2	10.0%	20.0%	0.0%	20.0%	5.0%	20.0%
		σ_x	20.0%	20.0%	20.0%	30.0%	**20.0%**	**25.0%**
		μ_y	70.0%	70.0%	70.0%	65.0%	**70.0%**	**67.5%**
		σ_y^2	0.0%	30.0%	20.0%	20.0%	10.0%	**25.0%**
		σ_y	60.0%	40.0%	40.0%	30.0%	**50.0%**	**35.0%**
		μ_z	40.0%	40.0%	50.0%	50.0%	**45.0%**	**45.0%**
		σ_z^2	0.0%	10.0%	0.0%	15.0%	0.0%	12.5%
		σ_z	60.0%	40.0%	40.0%	45.0%	**50.0%**	**42.5%**
		zcr_m	90.0%	60.0%	90.0%	75.0%	**90.0%**	**67.5%**
		$I_{max_{corr}}$	100.0%	80.0%	100.0%	75.0%	**100.0%**	**77.5%**
		T_m	60.0%	70.0%	80.0%	65.0%	**70.0%**	**67.5%**
		F_{c_1}	0.0%	60.0%	0.0%	55.0%	0.0%	**57.5%**
		F_{c_2}	20.0%	20.0%	20.0%	25.0%	**20.0%**	**22.5%**
		F_{c_3}	20.0%	30.0%	40.0%	30.0%	**30.0%**	**30.0%**
		F_{c_4}	0.0%	20.0%	20.0%	25.0%	10.0%	**22.5%**
		F_{c_5}	20.0%	60.0%	50.0%	45.0%	**35.0%**	**52.5%**
		F_{c_6}	10.0%	10.0%	30.0%	15.0%	**20.0%**	12.5%
		F_{c_7}	20.0%	60.0%	20.0%	55.0%	**20.0%**	**57.5%**
		F_{c_8}	30.0%	60.0%	30.0%	50.0%	**30.0%**	**55.0%**
		F_{c_9}	20.0%	20.0%	0.0%	15.0%	10.0%	17.5%
		$F_{c_{10}}$	20.0%	30.0%	20.0%	40.0%	**20.0%**	**35.0%**

REFERENCES

1. Shoaib, M., S. Bosch, O. D. Incel, H. Scholten and P. J. Havinga, "A Survey of Online Activity Recognition Using Mobile Phones", *Sensors*, Vol. 15, No. 1, pp. 2059–2085, 2015.

2. MathWorks, *Android Sensor support from MATLAB (R2013a, R2013b)*, 2015, http://www.mathworks.com/matlabcentral/fileexchange/ 40876-android-sensor-support-from-matlab–r2013a–r2013b-, [Accessed January 2015].

3. Ferreira, D., *AWARE: A mobile context instrumentation middleware to collaboratively understand human behavior*, Ph.D. Thesis, University of Oulu, 2013.

4. Kawahara, Y., H. Kurasawa and H. Morikawa, "Recognizing User Context Using Mobile Handsets with Acceleration Sensors", *Portable Information Devices, 2007. PORTABLE07. IEEE International Conference on*, pp. 1–5, 2007.

5. Lara, O. and M. Labrador, "A Survey on Human Activity Recognition using Wearable Sensors", *Communications Surveys Tutorials, IEEE*, Vol. 15, No. 3, pp. 1192–1209, 2013.

6. Preece, S. J., J. Y. Goulermas, L. P. Kenney, D. Howard, K. Meijer and R. Crompton, "Activity Identification Using Body-Mounted Sensors—A Review of Classification Techniques", *Physiological Measurement*, Vol. 30, No. 4, p. R1, 2009.

7. Lockhart, J. W., T. Pulickal and G. M. Weiss, "Applications of Mobile Activity Recognition", *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, UbiComp '12, pp. 1054–1058, ACM, New York, NY, USA, 2012.

8. Anderson, I., J. Maitland, S. Sherwood, L. Barkhuus, M. Chalmers, M. Hall, B. Brown and H. Muller, "Shakra: Tracking and Sharing Daily Activity Levels with Unaugmented Mobile Phones", *Mobile Networks and Applications*, Vol. 12,

No. 2-3, pp. 185–199, 2007.

9. Krumm, J. and E. Horvitz, "LOCADIO: Inferring Motion and Location from Wi-Fi Signal Strengths", *Mobile and Ubiquitous Systems: Networking and Services, 2004. MOBIQUITOUS 2004. The First Annual International Conference on*, pp. 4–13, Aug 2004.

10. Reddy, S., M. Mun, J. Burke, D. Estrin, M. Hansen and M. Srivastava, "Using Mobile Phones to Determine Transportation Modes", *ACM Transactions on Sensor Networks*, Vol. 6, No. 2, pp. 1–27, 2010.

11. Lane, N. D., Y. Xu, H. Lu, S. Hu, T. Choudhury, A. T. Campbell and F. Zhao, "Enabling Large-Scale Human Activity Inference on Smartphones Using Community Similarity Networks (CSN)", *Proceedings of the 13th international conference on Ubiquitous computing*, pp. 355–364, ACM, 2011.

12. Miluzzo, E., N. D. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, S. B. Eisenman, X. Zheng and A. T. Campbell, "Sensing Meets Mobile Social Networks: The Design, Implementation and Evaluation of the CenceMe Application", *Proceedings of the 6th ACM conference on Embedded network sensor systems*, SenSys '08, pp. 337–350, ACM, New York, NY, USA, 2008.

13. Lane, N., E. Miluzzo, H. Lu, D. Peebles, T. Choudhury and A. Campbell, "A Survey of Mobile Phone Sensing", *Communications Magazine, IEEE*, Vol. 48, No. 9, pp. 140 –150, 2010.

14. Incel, O., M. Kose and C. Ersoy, "A Review and Taxonomy of Activity Recognition on Mobile Phones", *BioNanoScience*, Vol. 3, No. 2, pp. 145–171, 2013.

15. Miluzzo, E., C. T. Cornelius, A. Ramaswamy, T. Choudhury, Z. Liu and A. T. Campbell, "Darwin Phones: The Evolution of Sensing and Inference on Mobile Phones", *Proceedings of the 8th international conference on Mobile systems, applications, and services*, MobiSys '10, pp. 5–20, ACM, New York, NY, USA, 2010.

16. Thiemjarus, S., A. Henpraserttae and S. Marukatat, "A Study on Instance-Based Learning with Reduced Training Prototypes for Device-Context-Independent Activity Recognition on a Mobile Phone", *IEEE International Conference on Body Sensor Networks*, pp. 1–6, 2013.

17. Park, J., A. Patel, D. Curtis, J. Ledlie and S. Teller, "Online Pose Classification and Walking Speed Estimation using Handheld Devices", *Proc. 14th International Conference on Ubiquitous Computing (UbiComp 2012)*, pp. 113–122, 2012.

18. Martin, H., A. M. Bernardos, J. Iglesias and J. R. Casar, "Activity Logging Using Lightweight Classification Techniques in Mobile Devices", *Personal and Ubiquitous Computing*, Vol. 17, No. 4, pp. 675–695, 2013.

19. Sun, L., D. Zhang, B. Li, B. Guo and S. Li, "Activity Recognition on an Accelerometer Embedded Mobile Phone with Varying Positions and Orientations", *Ubiquitous Intelligence and Computing*, Vol. 6406 of *Lecture Notes in Computer Science*, pp. 548–562, 2010.

20. Antos, S. A., M. V. Albert and K. P. Kording, "Hand, Belt, Pocket or Bag: Practical Activity Tracking with Mobile Phones", *Journal of Neuroscience Methods*, Vol. 231, No. Complete, pp. 22–30, 2014.

21. Yang, J., E. Munguia-Tapia and S. Gibbs, "Efficient In-Pocket Detection with Mobile Phones", *Proceedings of the 2013 ACM conference on Pervasive and ubiquitous computing adjunct publication*, pp. 31–34, ACM, 2013.

22. Lu, H., J. Yang, Z. Liu, N. D. Lane, T. Choudhury and A. T. Campbell, "The Jigsaw Continuous Sensing Engine for Mobile Phone Applications", *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, SenSys '10, p. 71–84, ACM, New York, NY, USA, 2010.

23. Google, *Sensor Manager Tutorial*, 2015, http://developer.android.com, [Accessed January 2015].

24. Avci, A., S. Bosch, M. Marin-Perianu, R. Marin-Perianu and P. Havinga, "Activity Recognition Using Inertial Sensing for Healthcare, Wellbeing and Sports Applications: A Survey", *23th International Conference on Architecture of Computing Systems, ARCS 2010*, pp. 167–176, 2010.

25. Eagle, N. and A. (Sandy) Pentland, "Reality Mining: Sensing Complex Social Systems", *Personal Ubiquitous Comput.*, Vol. 10, No. 4, pp. 255–268, 2006.

26. Mun, E. D. B. J., M. and M. Hansen, "Parsimonious Mobility Classification Using GSM and WiFi Traces", *Proceedings of the 5th Workshop on Embedded Networked Sensors (HotEmNets).*, 2008.

27. Ryder, J., B. Longstaff, S. Reddy and D. Estrin, "Ambulation: A Tool for Monitoring Mobility Patterns over Time Using Mobile Phones", *Proceedings of the 2009 International Conference on Computational Science and Engineering - Volume 04*, CSE '09, pp. 927–931, IEEE Computer Society, Washington, DC, USA, 2009.

28. Wu, W., S. Dasgupta, E. E. Ramirez, C. Peterson and G. J. Norman, "Classification Accuracies of Physical Activities Using Smartphone Motion Sensors", *Journal of medical Internet research*, Vol. 14, No. 5, p. e130, 2012.

29. Anjum, A. and M. Ilyas, "Activity Recognition Using Smartphone Sensors", *2013 IEEE Consumer Communications and Networking Conference (CCNC)*, pp. 914–919, Jan. 2013.

30. Yong, C. Y., R. Sudirman, N. H. Mahmood and K. M. Chew, "Human Hand Movement Analysis Using Principle Component Analysis Classifier", *Applied Mechanics and Materials*, Vol. 284, pp. 3126–3130, Trans Tech Publ, 2013.

31. Shoaib, M., S. Bosch, O. D. Incel, H. Scholten and P. J. Havinga, "Fusion of Smartphone Motion Sensors for Physical Activity Recognition", *Sensors*, Vol. 14, No. 6, pp. 10146–10176, 2014.

32. Liang, Y., X. Zhou, Z. Yu and B. Guo, "Energy-Efficient Motion Related Activity Recognition on Mobile Devices for Pervasive Healthcare", *Mobile Networks and Applications*, Vol. 19, No. 3, pp. 303–317, 2014.

33. Su, X., H. Tong and P. Ji, "Activity Recognition with Smartphone Sensors", *Tsinghua Science and Technology*, Vol. 19, No. 3, pp. 235–249, June 2014.

34. Siirtola, P. and J. Roning, "Ready-to-Use Activity Recognition for Smartphones", *2013 IEEE Symposium on Computational Intelligence and Data Mining (CIDM)*, pp. 59–64, Apr. 2013.

35. Siirtola, P., "Recognizing Human Activities User-independently on Smartphones Based on Accelerometer Data", *International Journal of Interactive Multimedia and Artificial Intelligence*, Vol. 1, No. Special Issue on Distributed Computing and Artificial Intelligence, pp. 38–45, 2012.

36. Guiry, J. J., P. v. d. Ven and J. Nelson, "Orientation Independent Human Mobility Monitoring with an Android Smartphone", ACTAPRESS, 2012.

37. Blanke, U. and B. Schiele, "Sensing Location in The Pocket", *Ubicomp Poster Session*, 2008.

38. Yang, J., "Toward Physical Activity Diary: Motion Recognition Using Simple Acceleration Features with Mobile Phones", *Proceedings of the 1st International Workshop on Interactive Multimedia for Consumer Electronics*, IMCE '09, pp. 1–10, ACM, New York, NY, USA, 2009.

39. Ustev, Y. E., O. Durmaz Incel and C. Ersoy, "User, Device and Orientation Independent Human Activity Recognition on Mobile Phones: Challenges and a Proposal", *Proceedings of the 2013 ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication*, UbiComp '13 Adjunct, pp. 1427–1436, ACM, 2013.

40. Berchtold, M., M. Budde, D. Gordon, H. Schmidtke and M. Beigl, "ActiServ: Activity Recognition Service for Mobile Phones", *2010 International Symposium on Wearable Computers (ISWC)*, pp. 1–8, Oct. 2010.

41. Henpraserttae, A., S. Thiemjarus and S. Marukatat, "Accurate Activity Recognition Using a Mobile Phone Regardless of Device Orientation and Location", *Proceedings of the 2011 International Conference on Body Sensor Networks*, BSN '11, pp. 41–46, IEEE Computer Society, Washington, DC, USA, 2011.

42. Miluzzo, E., M. Papandrea, N. Lane, H. Lu and A. Campbell, "Pocket, Bag, Hand, etc.-Automatically Detecting Phone Context Through Discovery", *Proc. of the First International Workshop on Sensing for App Phones (PhoneSense'10)*, 2010.

43. Dogan, E., *Efficient Feature Selection for Online Activity Recognition on Smart Phones*, M.S. Thesis, Bogazici University, 2013.

44. Kose, M., O. D. Incel and C. Ersoy, "Online Human Activity Recognition on Smart Phones", *Workshop on Mobile Sensing: From Smartphones and Wearables to Big Data*, pp. 11–15, 2012.