To Design and Implement a Smartphone Based Dynamic Fall Detection System

This thesis is submitted in partial fulfillment of the requirement for the degree of Bachelor of Science in Computer Science and Engineering.

A S M Nazrul Islam
ID: 1204021

Supervised by
Abu Hasnat Mohammad Ashfak Habib
Assistant Professor
Department of Computer Science and Engineering (CSE)
Chittagong University of Engineering and Technology (CUET)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
CHITTAGONG UNIVERSITY OF ENGINEERING AND TECHNOLOGY (CUET)
CHITTAGONG – 4349, BANGLADESH.

The thesis titled "**To Design and Implement a Smartphone Based Dynamic Fall Detection System**" submitted by ID 1204021, session 2015-2016 has been accepted as satisfactory in fulfillment of the requirement for the degree of Bachelor of Science in Computer Science and Engineering (CSE) as B.Sc. Engineering to be awarded by Chittagong University of Engineering and Technology (CUET).

Board of Examiners

1. ______________________________ Chairman

Abu Hasnat Mohammad Ashfak Habib
Assistant Professor
Department of Computer Science and Engineering (CSE)
Chittagong University of Engineering and Technology (CUET)

2. ______________________________ Member (Ex-officio)

Dr. Mohammad Shamsul Arefin
Professor and Head
Department of Computer Science and Engineering (CSE)
Chittagong University of Engineering and Technology (CUET)

3. ______________________________ Member (External)

Dr. Asaduzzaman
Professor
Department of Computer Science and Engineering (CSE)
Chittagong University of Engineering and Technology (CUET)

Statement of Originality

It is hereby declared that the contents of this project is original and any part of it has not been submitted elsewhere for the award if any degree or diploma.

Signature of the Supervisor	Signature of the Candidate
Date:	Date:

Acknowledgement

Prima facie, I am grateful to the Almighty for giving me the strength for successful completion of this project. Then I would like to express my sincere gratitude to my honorable project supervisor Abu Hasnat Mohammad Ashfak Habib, Assistant Professor, Department of Computer Science and Engineering, Chittagong University of Engineering and Technology, for his valuable advices, constructive suggestions and sincere guidance with all the necessary facilities for assimilation, research and preparation for the project. I place on record, my sincere gratitude to Dr. Asaduzzaman, Professor, Department of Computer Science and Engineering, Chittagong University of Engineering and Technology, for his kind encouragement and cooperation. I would like to thank my family for their constant love and support. Finally, I would like to take this opportunity to express my gratitude to one and all, who directly or indirectly, have lent their hand in this venture.

Abstract

A dynamic fall detection system is a key issue in ensuring sound daily life, reducing fall related injuries and reducing post fall affects. Based on only smartphones in-built sensors we have designed and developed an effective dynamic fall detection system. The system will detect fall for any smartphone position and will place an instant communication with the care-giver with the victim's location.

We have detected fall using only smartphone, no external sensors or gadgets were used in our fall detection system. Smartphone's in-built sensors were only used to detect fall. We estimated fall by monitoring user's movement and surrounding environment with the help of smartphone's inbuilt sensors like- accelerometer sensor, orientation sensor and microphone sensor. We have detected fall, based on these sensors real time data. By calculating user's total acceleration and sound level of the environment from the sensor's real time data, we have developed our system. Each of the parameter dictates relevant information about the user's present condition. These parameters information then merged to get the accuracy of the fall decision. Depending on the values the system detects human fall and tries to get the user's attention by generating an alarm. Upon failing to get the user's attention the system will automatically and instantly notify the care-giver with the user's location for quick medical aid. And statistics show that the majority of serious consequences are not direct result of fall rather it is due to delay in aid, treatment or in worst case detection.

Though real life experiment with real persons was not possible, but the system was experimented using a mannequin and two human shaped scarecrow dummy of different height. Testing objects were used to mimic the human fall. The test results were promising. More than 70 falls were mimicked placing smartphone either in shirt's pocket and pants pocket. The test results showed us that the developed system is functioning well and attaining high accuracy in detecting fall. And the rate of false negativity is also low with sensitivity of 89.04% and specificity of 75%. Our proposed system with smartphone's in-built multiple sensor fusion and their classified results can be an exquisite candidate in adaptive, user-oriented health-care system.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Falls are considered as the second leading cause of accidental or unintentional injury deaths worldwide. A fall is defined as an event which results in a person coming to rest inadvertently on the ground or floor or other lower level [1]. Fall-related injuries may be fatal or non-fatal. For example, in People's Republic of China, for every death due to fall, there are 4 cases of permanent disability, 13 cases requiring hospitalization for more than 10 days, 24 cases requiring hospitalized for 1-9 days and 690 cases seeking medical care or missing work/school [1]. The largest morbidity occurs in people aged 65 years of older, young adults, aged 15-29 years and children aged 15 years of younger.

Each year around 6, 46,000 individuals [1] die globally from falls. In previous year the statistics were 4, 24,000 individuals (according to WHO). Regions of Western Pacific and South East Asia are accounting for 60% of these deaths. Aside from death, falls are also responsible for over 17 million DALYs (disability-adjusted life yeas) lost [1]. While, nearly 40% of the total DALYs lost due to falls worldwide occurs among children. Around 37.3 million falls are severe enough to require medical attention. One out of five falls causes a serious injury like broken bones or head injury [2, 3]. More than 95% of hip fractures are caused by falling [4]. Falls are also most common cause of traumatic brain injuries [5] and 80% of this occurs in low and middle-income countries.

Individuals from all age group are prone to fall, starting from children to old. And this fall rate is high especially among children, pregnant-women and old men. Though, the risk of fall increases below 6 years and above 60 years gradually. In the year 2004, nearly 47,000 children and young under the age of 20 years died [6] as a result of fall.

Children falls as a result of their evolving developmental stages, innate curiosity of their surroundings and the alarming increasing levels of independence that coincide with more challenging behaviors commonly referred to as "risk factor". In adequate adult supervision, interacting with poverty, sole parenthood is also considered as risk factor. According to WHO, in Brazil, for children under 15 years of age, more than half of all non-fatal injuries are the results of fall [6]. In America, fall is 1.1% of total DALY's, in South-east Asia Regions, it is 1.7%. In European regions it is 2.2%. In Bangladesh also the lead cause of child-injury is fall. The fatal fall injury rate was 2.8/100 000 for children of all ages. The percentage of mortality by fall for boys of age less than 1 year is 8.4. While among girls of less than 1 year it is 36%. Around 770 children are injured each day. Among them 10 are cursed with permanently disability each day [7].

In case of pregnant-women, due to their ongoing physical transformation, they are very prone to sudden seizure and loosing sense. Failure of immediate assistance can affect both mother and child.

Old peoples are much more prone to falls. Each year 2.8 million older people are treated in emergency departments for fall injuries [8]. In United States of America, 20-30% of older people who fall suffer moderate to severe injuries such as bruises, hip fractures or head traumas. It has been observed that males are more likely to die from a fall, while female suffers more non-fatal falls. Worldwide males consistently sustain higher death rates and DALYs lost.

Financial costs from fall-related injuries are substantial. For people aged 65 years or older, the average health system cost per fall injury in the Republic of Finland and Australia are US$ 3611 and US $ 1049 respectively [1]. Direct medical costs for fall in uries are $ 31 billion annually [9]. Hospital costs account for two-thirds of the total.

Additionally, many people who fell, even not being injured, become afraid of falling. This fear may cause a person to shrink their everyday activities. When a person is less active they became weaker and this increases their chances of falling [10].

So, the considerable risks of falls and the substantial increase in population has stimulated both commercial product and academic research on fall detection and prevention. A typical fall detection system complies of two major parts: the detection component which detects fall and the communication component which communicates with the given contact persons. Evidence from China suggests that the implementation of effective fall prevention would create a net savings over US $ 120 million each year.

1.1 Background and Present State of the Problem

Due to different adverse effects in social, physical, psychological and economical, falls have been an attractive issue to the researchers for the last decade. Fall detection techniques can be generalized into three categories: sensor-based detection, database-based motion type classification and visual-based detection.

1.1.1 Sensor Based Fall Detection Process

In sensor-based detection process, in general external sensor's value like, accelerometer sensor's value is used. And when the accelerometer sensor is used most methods are based on thresholds. In [11], Nyan et al. used an accelerometer that has to be settled in the shoulder garment. Kangas et al. in [12] proposed using four thresholds for total sum vector, dynamic sum vector, vertical acceleration and difference between the acceleration's maximum and minimum values. Other than accelerometer, gyroscope sensor's value is also used in fall detection. In [13, 14] information of body's orientation and pressure are also used to detect acceleration based fall detection. Pannurat et al. in [15] offered a commercial off-the-shelf wearable device for fall detection. The most common position for these devices is waist position provided with a 'panic button'. And most products are powered by a lithium-ion battery and employ embedded sensors to detect falls. CARA architecture [16] proposed to send the data of a wearable accelerometer to an external non-portable gateway which applies a threshold method to detect fall. ZigBee is the wireless technology between the wearable sensors and a central server in the system that is portrayed in [17]. Arduino Fio hardware containing a gyroscope and a accelerometer is used as a compact to detect falls in [18] by sending an alarm via Bluetooth to a nearby smartphone.

Though recently, due to the availability of multiple embedded sensors and cost shrinking, smartphones have got the full attention and attraction. But, in previous and still external circuit/device is being used to detect falls. Brickhouse [19] consists of a tele-assist base and a portable sensor. In which base device is needed to be installed indoors. So, the signal transmission distance between the base and the sensor is limited and results in severe drawbacks. Like, the monitoring place is fixed and the installment, adjustment and monitoring cost are quite high. Medical Guardian [20] provided an economic fall alarm monitor that has a fixed distance range and works by using nation-wide cellular network but, it has been awarded ‘Gold Award’ for the year 2016 for its customer review.

1.1.2 Database Based Fall Detection Systems

In case of database based fall detection systems, Ganti et al. in [21] and Karantonis et al. in [22] proposed for storing sensed user behavior data into a database for various activities like, fall down and recognition hereafter. These databases stored with sensed data are very useful. They can also be used for detecting much more activities and for pattern recognition of an individual. Fu et al. in [23], Sixsmith et al. in [24], Miaou et al. in [25] and Jansen et al. [26] proposed for capturing images of people and then detecting visual falls based on image processing techniques.

1.1.3 Smartphone Based Fall Detection Systems

But in last few years, the number of fall detection apps in smartphones is increasing day by day due to easy availability of smartphones and multiple sensors embedded in them. And these are being developed in various operating systems of smartphones. Like, initial smartphone based fall detection were developed in Symbian OS on Nokia phones [27]. In [28], a Java multiplatform software architecture, using external architecture was employed in both a Symbian phone (Nokia 5800) and Android smartphones (Samsung Galaxy, HTC hero) to detect falls.

In android platform, a vast literature on fall detection systems has been generated. The general principles of fall detection for elderly have been summarized by Yu [29] and Noury [30] while different classifications of fall assessment techniques have been presented by Pannurat et al. [15], Hedge et al. [31], Noury et al. [30, 32], Hijaz et al. [33]. However, smartphone-based fall detection and prevention architectures are specifically revised in a great detail by Habib et al. in [34].

But, in the prevailing systems there exist some obstacles like; the smartphone’s position is kept fixed. So, a fixed threshold had to use. But it is inconvenient in real life. Because an individual can keep his/her smartphone in any place in accordance with his/her comfort. So, the existing techniques are not dynamic. Hence, my approach will be to eradicate this limitation.

1.2 Motivation

A significant amount of death and fatal injuries are happening throughout the world for intentional and unintentional fall. And as the day passes the mortality rate due to fall is increasing at an alarming rate. Though there are few systems at present available for fall detection, but due to some constraints, they aren’t the optimal one. So, the aims of the proposed and implemented system are:

- To design and implement a novel smartphone-based fall detection technique that uses multiple numbers of inbuilt smartphone sensors.
- To make the technique dynamic so that it can automatically detect the position of the smartphone and select the appropriate threshold values accordingly.
- To make the technique false alarm exempt by fusing multiple sensor’s value, so that it can avert the limitations of the previous works.
- To make the technique robust and efficient by placing instant communication right after the occurrence of fall. Hence, resulting in restoring life.

So, the main objective of this project is to design and implement a reliable and dynamic fall detection system, which has low fault tolerance and has higher efficiency.

1.3 Contributions

The goal of this project is to design a smartphone based dynamic fall detection system and develop it. Persons of any age range can run this system on his/her smartphone regardless of whichever smartphone he/she has and can be benefited. Especially the persons who live in isolation or of their own like elders, pregnant women will be benefited more.

1.4 Organization of the Paper

The following chapters will go through the different aspects of this project. Chapter 1 gives exordial concept of our project. Chapter 2 gives, an overview of our project related terminologies and brief discussion of the previous works with their limitations. Chapter 3 describes the working procedure of our project. In Chapter 4, we have illustrated our implementation of the project in details. Chapter 5 centers on the experimental result and evolution of the proposed system. The paper concludes with a summary of our work and future recommendations for further improvements in Chapter 6. This paper contains one appendix, intended for persons who wish to explore certain topics in greater depth. Appendix A contains a fragment of the source code of the project.

Chapter 2

Literature Review

In this chapter, we present studies on the terminologies related to the project which are important to understand. This chapter also contains brief discussion on related previous works.

2.1 Introduction

The smartphone-based dynamic fall detection system is a system that is developed to detect fall. The system can detect fall for all range of persons. Due to the advancement of technology at present smartphone is a must and has become a part and parcel of our daily activities. It has become an inseparable part of our daily life. So, to reach out to the maximum number of individuals, the proposed system has been developed in the most demandable and useable platform- android platform. Now-a-days smartphone is also playing a vital role in mutual communication. And this feature is used to reach out the near ones in times of fall. There are some systems as applications to detect fall. But due to some constraints, they are not the optimal one. To eradicate those limitations the proposed system is implemented as an application. This application will use the smartphone to dynamically monitor for fall while the smartphone can also be used as per user need. Unintentional death rates, due to fall in United States is shown in figure 2.1.

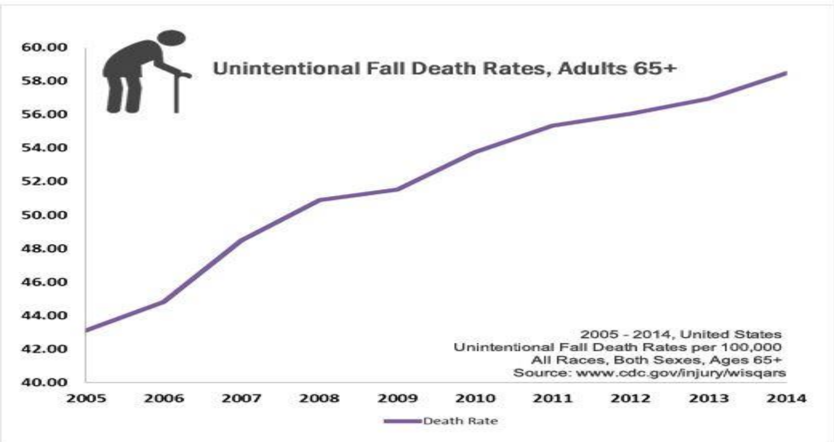

Figure 2.1: Chart for unintentional death rates due to fall

2.2 Fall

Fall is defined as the event which results in a person coming to rest inadvertently on the ground or floor or other lower level [1]. It can be intentional or unintentional. But the result of this is unavoidable. It can be fatal or non-fatal. Fall is considered as the leading cause of injury deaths and of injury-related hospitalization. If treatment is not obtained within a short period of time, internal injuries may also lead to a complex deadly condition.

All human experience falls in their life. But infant and elderly peoples are more prone to fall. And if they are not properly and timely aided fall-related accidents lead to serious consequences in their case. Besides them, due to occupation, people working in high places, tall buildings/ skyscrapers like construction workers, window washers, painters and even mountaineers are frequent victim of fall. An elderly human fall is pictured in figure 2.2.

Statistics show that the majority of serious consequences are not direct result of fall rather it is due to delay in aid, treatment or in worst case detection.

Figure 2.2: Human fall

2.3 Stages of Fall

During falling, human body experiences a variety of change of states. Depending on these states a falling event can be catheterized. These are identified in below figure 2.3 and described in below section:

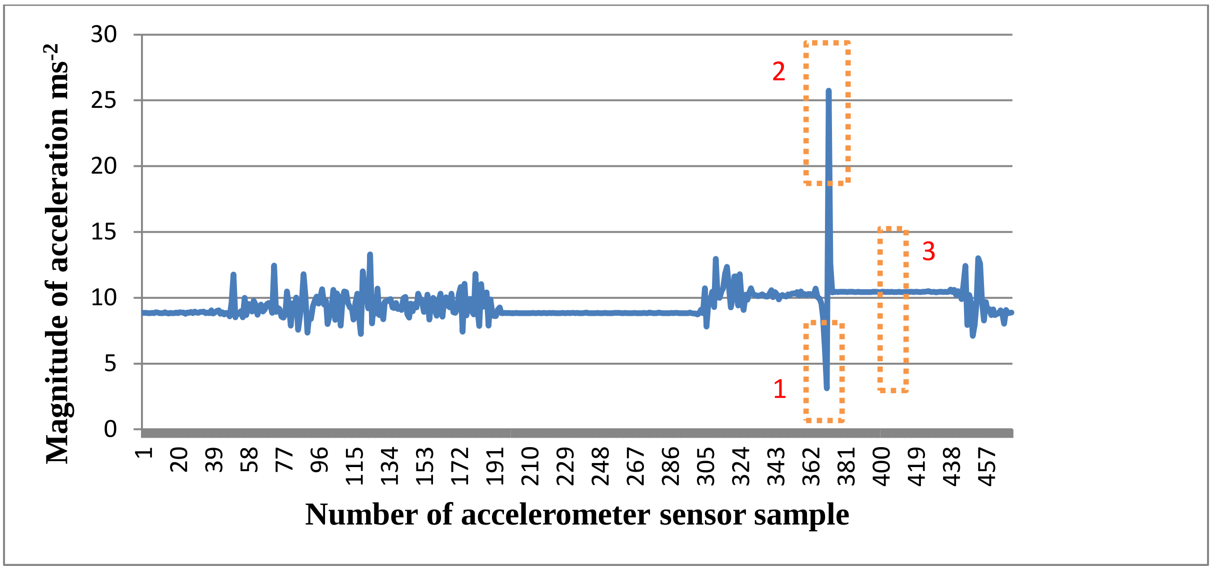

Figure 2.3: Changes of acceleration due to a fall event

1. **Initiation:**

This is the first stage of the fall. A feeling and phenomenon of weightlessness will be felt at the initial stage of fall. At this stage, the tri-axial sum of acceleration i.e. total acceleration of the falling object tends towards 0 g. Free fall and falling from a great height results in total acceleration almost equal to 0. But in case of human fall phenomenon, the value of the tri-axial sum of acceleration does not become 0 or tends to 0. It degrades but doesn't tend to be equal to 0. But still, it will be less than 1g, which occurs rarely in normal condition. And the duration of this stage depends on the height of the fall. For greater height, the duration is much larger than the fall having less height. So, this is considered as the pre-condition for considering a fall.

2. **Meting Ground:**

    This stage comes right after experiencing weightlessness. It is for sure that for each fall the falling object will meet the ground due to gravitational force. And this ground meeting stage has a great impact on the total acceleration of the falling object.

3. **After Meeting Ground:**

    The post experience of after meeting the ground varies according to the type of the falling object and falling place. In case of inert or light object right after impact, the object generates a vertical acceleration as a result of reaction, due to which it rises upward. And this rise or upward movement also depends on the mass and height of the fall. But in case of human fall, the human body doesn't experience this phenomenon. Right after experiencing weightlessness and ground impact, a human body can't rise immediately. Rather it stays in a motionless condition for a certain time.

2.4 Types of Fall

Depending on the position and orientation of the falling body there is a classification of fall. They are-

2.4.1 Forward Fall

This indicates the falling condition in which fall occurs in the forward direction. The subject falls, facing down. Pictorial view is given in figure 2.4.

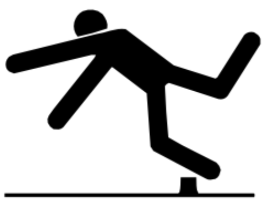

Figure 2.4: Froward fall

2.4.2 Backward Fall

This indicates the falling condition in which fall occurs in the backward direction. That is the subject falls facing up. Pictorial view is given in figure 2.5.

Figure 2.5: Backward fall

2.4.3 Lateral Fall

This indicates the falling condition in which fall occurs in the lateral direction. That is the subject falls either on its right or left. Pictorial view is given in figure 2.6.

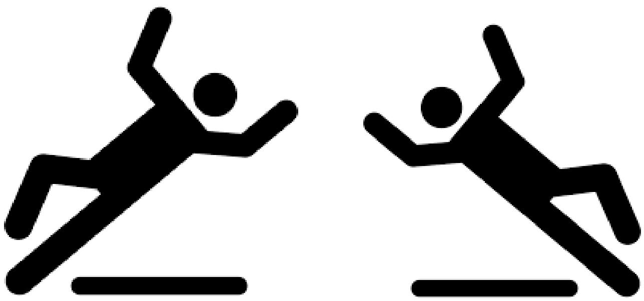

Figure 2.6: Lateral fall

2.5 Causes of Fall

Though fall is considered as a normal event of our life, it can be caused by a number of risk factors which can affect us severely resulting in hindering to carry out our daily activities. And the risk factors are those that can increase the chance of developing a problem, hazard or disease. Risk factors can be personal, related to our body, habit or even lifestyles or environmental like slippery road or stairs, dizziness due to excessive heat. But whatever may be the type, the risk factors affect our body balance and daily activities.

2.5.1 Physical Risk Factors [36]

- Aging – with aging human body gradually goes out of control. The immune system becomes weak. So, the mind and body both get disturbed or fragile.
- Anemia or other blood disorders – due to these blood disorders human body can't get enough supply of energy to fulfill the requirements for secure metabolism
- Thyroid problems
- Foot disorders – problems with an individual's feet like – bunions, ingrown or thick nails, ulcerations or even uncomfortable footwear can also increase an individual's risk of falling.
- Muscle weakness- with aging human body gets weaker day by day resulting in loss of strength and balance, which can lead to a fall.
- Vertigo of balance difficulties
- Sensory disorders such as vision or hearing
- Brain or mood disorders including dementia, Alzheimer's disease, depression can
- Urinary incontinence or having to urinate at a frequent frequency. Cause going bathroom frequently in a rush increases the possibility of fall.
- Dehydration – lack of fluids in the body. While aging our body loses water. And lose of water causes dehydration. Which results in hypotension (low blood pressure), loss of balance, constipation and many other unwelcomed symptoms, which can bring on a fall.
- Psychological – fear of fall. Previously fell person grows a fear of fall. This tendency is greater among senior persons.
- Low vitamin D – lack of vitamin D can also lead to a fall.
- Other factors are Arthritis, Chronic pain, Diabetes, Parkinson's disease.

Identification of references (total 87) for pathologies of aging that may cause falls is given in below figure 2.7 [35].

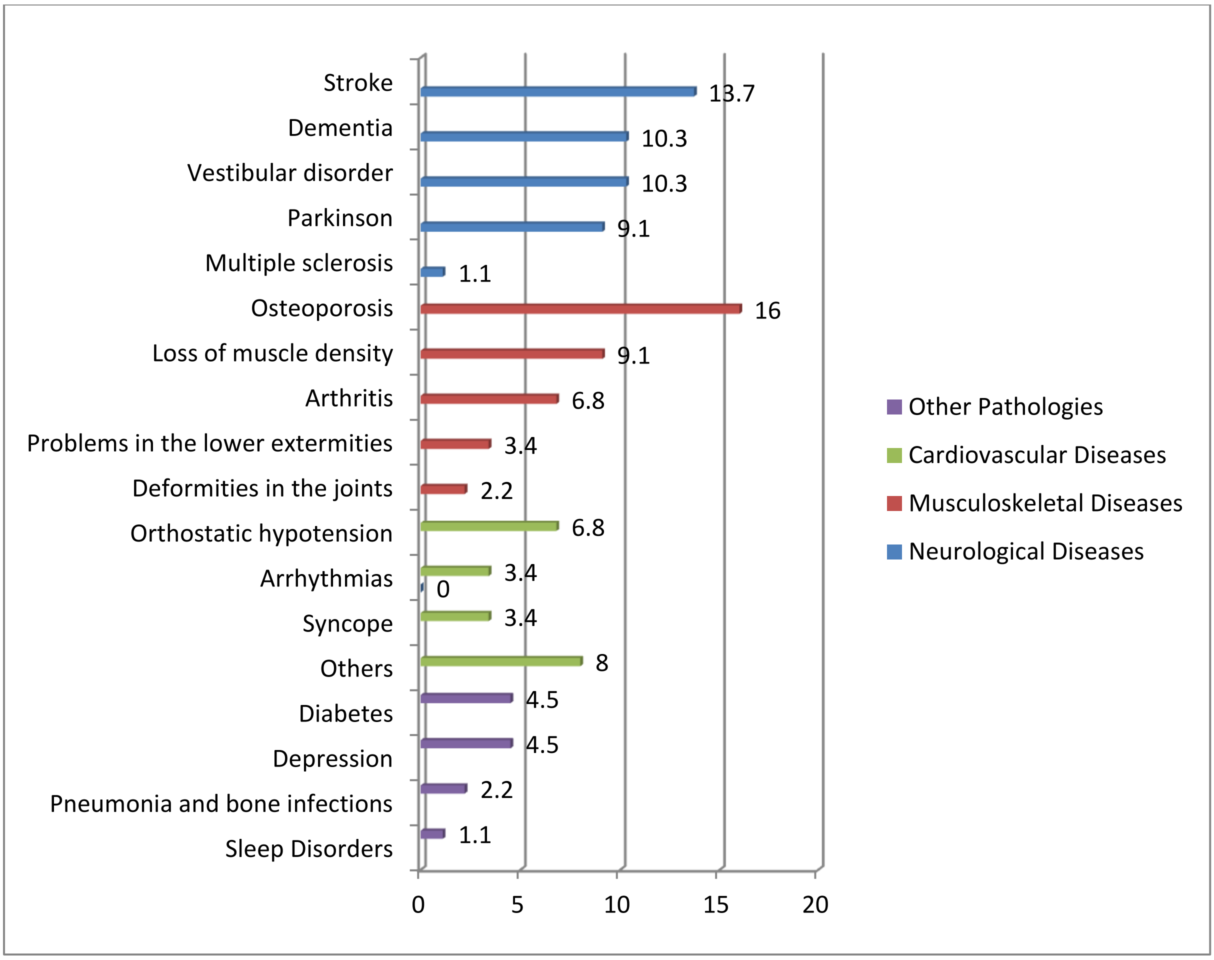

Figure 2.7: Identification of references (total 87) for pathologies of aging that may cause falls [35]

2.5.2 Environmental Factors

- Adverse weather - excessive heat or humidity causes sweating more than normal, which can lead to dehydration or dizziness resulting in fall.
- Improper footwear – wearing shoes with high heels or slippery works as a catalyst for fall.
- Natural risks – like uneven ground, slippery road.
- Risk inside of home – unorganized lifestyle, dark stair or corridor, loose carpet, wet floor or even lose wire can cause fall inside of a house.

2.6 Consequences of Fall

Though the consequences of a fall depend on the type and height of the fall, whatever is the type, every fall affects the victim either physically or psychologically. Fall has several post complexities. It is considered as the gateway of several health complications. The consequences of fall are described below in a pictorial way in figure 2.8, figure 2.9 figure 2.10 and figure 2.11. This information is obtained from [35].

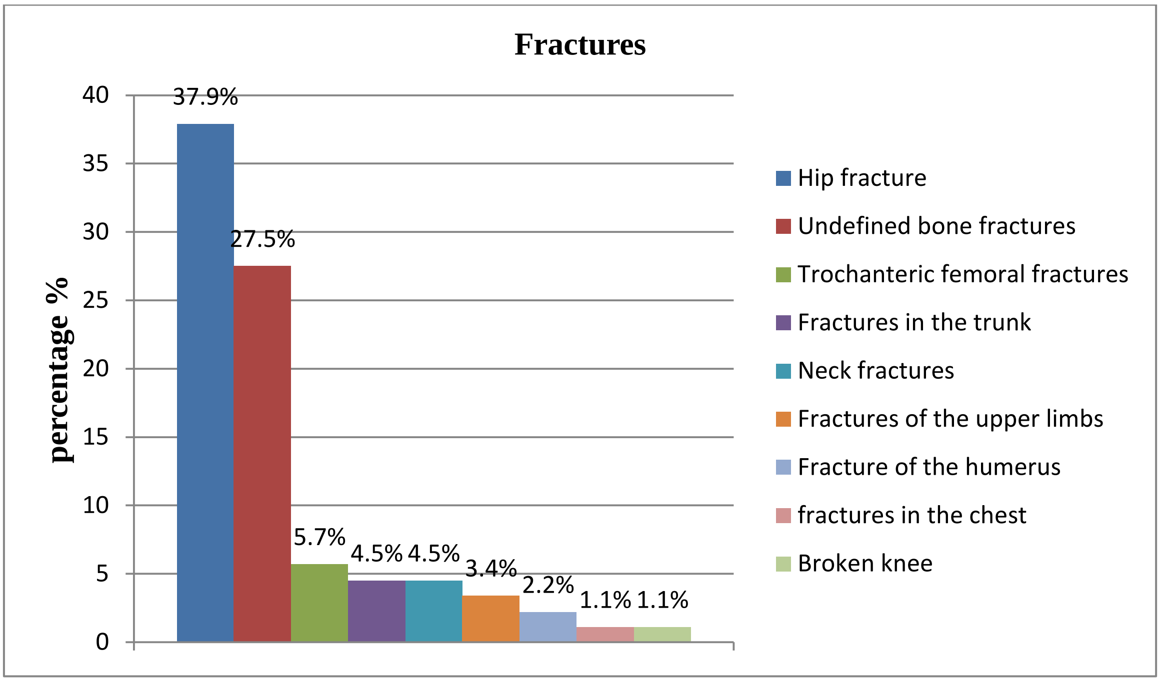

Figure 2.8: Percentage of fractures

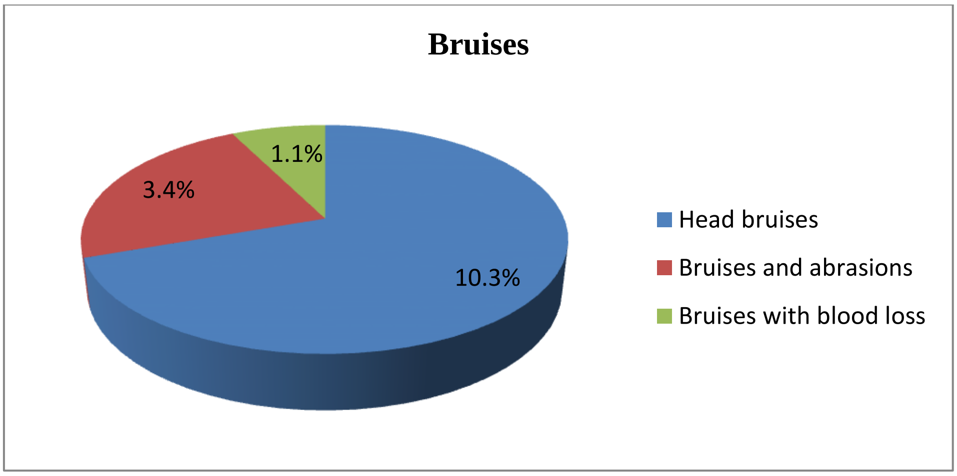

Figure 2.9: Percentage of bruises

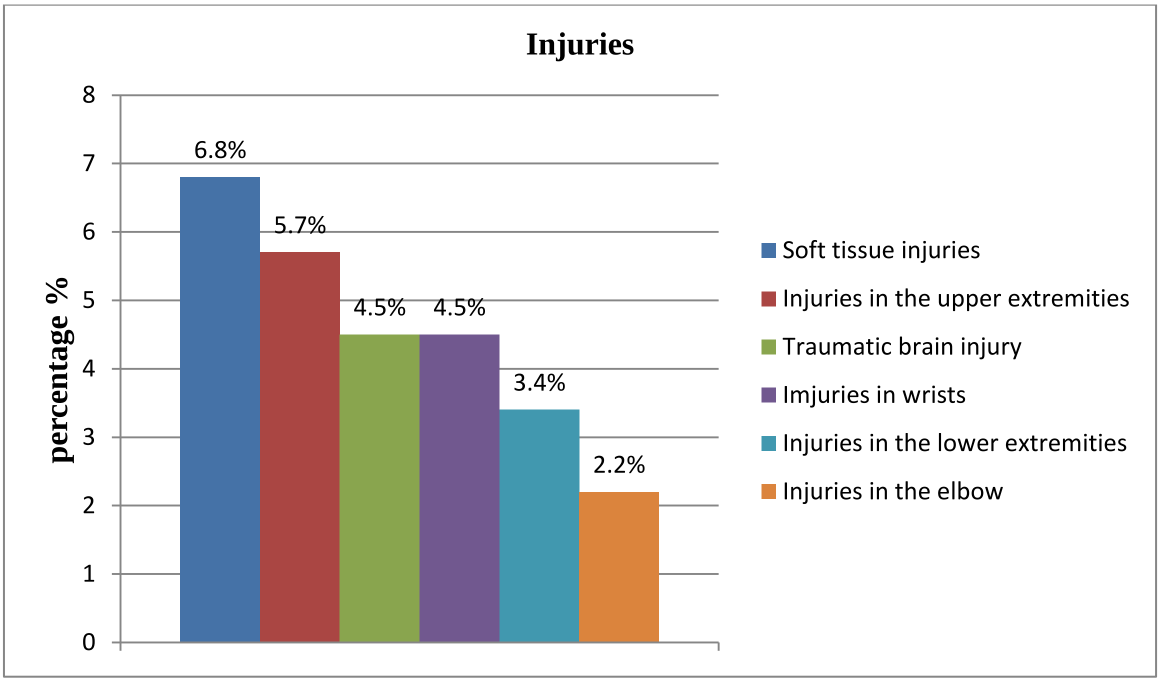

Figure 2.10: Percentage of injuries

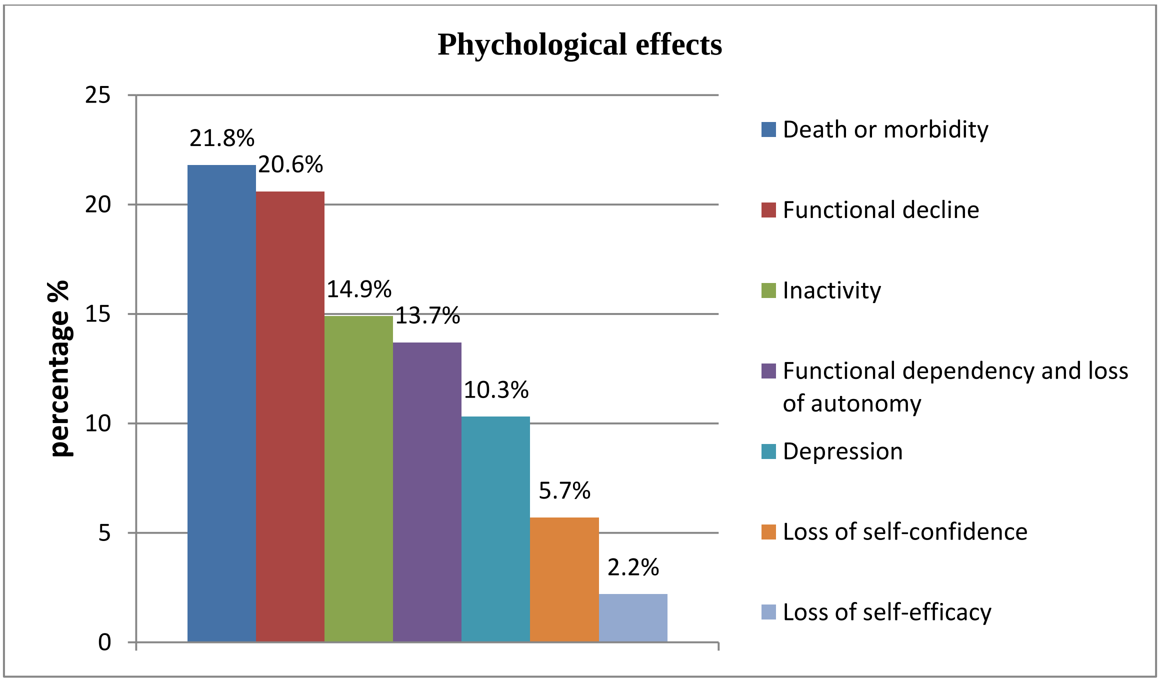

Figure 2.11: Percentage of psychological effects

2.7 An Overview of Android

Android is one of the most popular software platform and operating system for mobile devices which are touchable like smartphones, tablets. It was developed by Android Inc., which was later on bought by in 2005. But it was unveiled in 2007. It bases on the Linux Kernel. It is complete software stack that overrides all middleware needed to run the end-user application on mobile devices. Android is open source. Meaning, developers can modify and customize the OS as per need. Therefore though having same OS different mobile devices may have different graphical user interface GUI. Android supported mobile devices come with several built-in applications and also support third-party programs. Developers can create programs for Android platform using the Android SDK (Software Development Kit). Starting from September 2008 to August 2017 Android has gone through multiple releases. All Android mobile devices have a commonplace known as Google Play Store, in which and from which applications can be uploaded and downloaded as per need. At present, there are 2.7 million apps in Google Play Store. An android run smartphone is shown in figure 2.12.

To develop an Android application, Android SDK is a must. This SDK is a very comprehensive tool that contains not only the library for development but also the simulator to test the application. In general editors like Eclipse, Netbeans, and Android Studio are used to develop an application in this platform. The Android SDK provides the API libraries and developer tools necessary to build, test and debug android programs. Its touch gestures give users a feel like real-world actions- tapping, swiping.

Figure 2.12: An Android based smartphone

2.8 Android Platform Architecture

Android is an open source, Linux-based software stack that is created for a wide range of devices and form factors. There are some major components of the Android platform. They are discussed below:

2.8.1 Linux Kernel

Linux Kernel is the foundation of the Android platform. For functionalities like threading, low-level memory management Android Runtime (ART) relies on the Linux Kernel. Linux Kernel also allows Android to take the advantage of key security features. Thus Original Equipment Manufacturers can use Linux on their system and have the drivers running before loading other components of the stack. This is pictured in below figure 2.13.

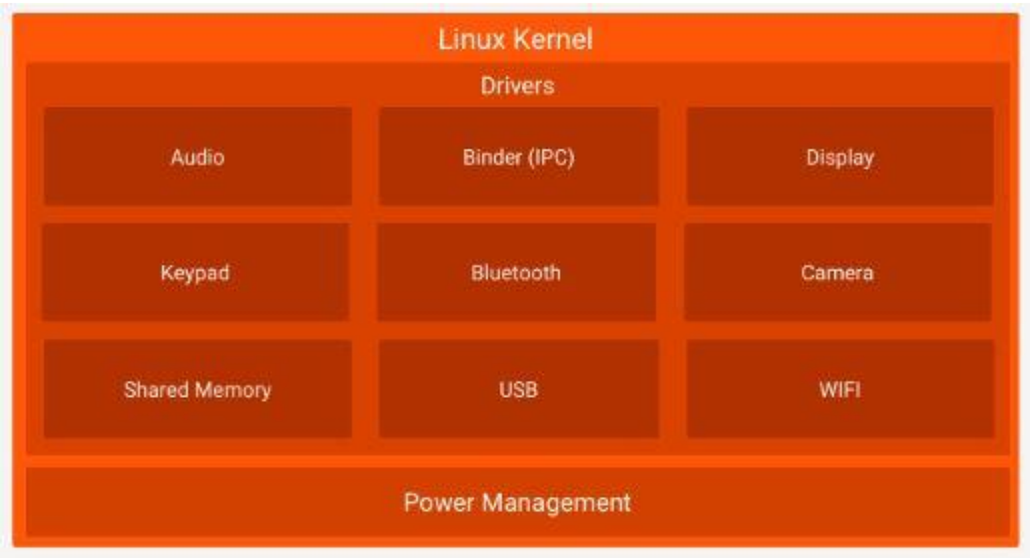

Figure 2.13: Linux Kernel

2.8.2 Hardware Abstraction Layer

The hardware abstraction Layer in short HAL provides standard interfaces that expose device hardware capabilities to the higher-level Java API framework. The HAL consists of multiple library modules like camera or Bluetooth module based on the type of hardware component to implement an interface. When a framework API makes a call to access device hardware, the Android system loads the library module for that hardware component. This is pictured in below figure 2.14.

Figure 2.14: Hardware abstraction layer

2.8.3 Android Runtime

Android Runtime in short ART is used to run multiple virtual machines on low-memory devices by executing DEX files, which is a byte-code format used for optimized minimal memory footprint.

ART can also do Ahead of time (AOT), just in time (JIT) compilation and optimized garbage collection (GC). It also gives better debugging support. This is pictured in below figure 2.15.

Figure 2.15: Android Runtime

2.8.4 Native C/C++ Libraries

The core Android system components like ART, HAL are built from native codes that require native libraries written in C/C++. The surface manager is responsible for composing, coordinating and rendering surfaces on the screen from windows owned by different applications, running in different applications, running in different processes. It manages access to the display subsystem and composite 2D and 3D graphics layers from multiple applications. For example, media libraries provide all of the audio and video codes accountable for rich media experience. Free type is used for managing and rendering fonts on the screen. This is pictured in below figure 2.16.

Figure 2.16: Native C/C++ libraries

2.8.5 Java API Framework

API’s forms a bridge between Android OS and developer. The API’s act as the building block to create an Android app by simplifying the core reuse, modular system components and services.

API includes a View system to build app’s UI, a resource manager to allow the access to the non-code resources, a notification manager to display custom alerts, an activity manager to manage the lifecycle of the apps and a content provider to enable the apps to access data from other apps. This is pictured in below figure 2.17.

Figure 2.17: Java API Framework

2.8.6 System Apps

It is the top-most layers. Android comes with a bunch of core apps for calling, messaging, internet browsing etc. The system apps works both as apps for the users and as a helper for the developers for their own app. This is pictured in below figure 2.18.

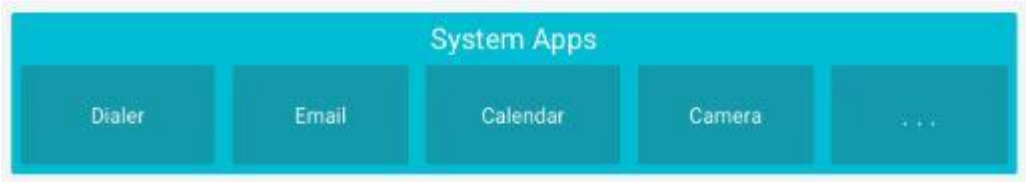

Figure 2.18: System apps

2.9 Related Work

Though first fall detection system was introduced by Hormann in early 1970s [37] but Hansen et al. [38] used the smartphone's camera for the first time in 2005 for fall detection.

Technique:

They used an external sensor based device to collect the sensor data and then passed these data via Bluetooth to a smartphone to analyze the data. If the user doesn't correspond to the system call, a photo is taken and sends to a central server accompanied via previous data.

Limitations:

- Have to wear the external device to detect fall
- Bluetooth connection works for a small range
- Much more prone to false negativity due to depending on only the accelerometer.

The prevailing fall detection systems can be classified into three categories based on their fall detection technique.

1. Accelerometer based fall detection
2. Database based fall detection
3. Image processing based fall detection

2.9.1 Accelerometer Based Fall Detection Systems

When Accelerometer sensor is used to detect fall, the most likely used methods are threshold based. Like-

2.9.1.1 Threshold Based System

In [11], Nyan et al. used accelerometer to detect fall. They set the accelerometer into garment on the shoulder position. The absolute peak values of acceleration were considered as the threshold. Based on the threshold, a fall is detected.

In [12], Kangas et al. proposed a system where four threshold values were considered. They were total sum vector, dynamic sum vector, vertical acceleration and difference between the maximum and minimum acceleration values. But according to their proposal, a fall is considered detected as long as one threshold is exceeded.

These threshold-based systems, in order to make the decision about a potential fall, compare the accelerometer sensor's output values with the predefined threshold values.

Advantages of Threshold-Based Fall Detection Systems

- The threshold based fall detection systems are less complex.
- They used less complex algorithms, so required lowest computational power.

Limitations of Threshold-Based Fall Detection Systems

- Most of the systems used predefined fixed threshold.
- Even if adaptive threshold values were used, it was not calculated dynamically while running the system.
- They are applicable for a predefined fixed place of human body condition.
- Much more prone to false negativity due to depending only in accelerometer's value.

2.9.2 Database Based Fall Detection System

In [21], Ganti et al. and in [22], Karantonis et al. proposed to store the sensed user behavioral data into a database for various daily activities like- walking, falling and use these afterward.

Advantages of Database Based Fall Detection Systems

- The built databases become handy for any later activity recognition.
- The stored data can be used to detect further any normal or abnormal activities.

Limitations of Database Based Fall Detection Systems

- The system's performance fully depends on the database.
- The system's performance may become inefficient if the database is compromised or undergo unauthorized modification.
- Maintaining a secured database may become challenging in sense of economics.

2.9.3 Image Processing Based Fall Detection System

In [23] Fu et al., in [24] Sixsmith et al., in [25] Miaou et al. and in [26] Jansen et al. proposed for capturing images of users and then detecting visual falls based on image processing techniques.

Advantages of Image Processing Based Fall Detection Systems

- A promising way to detect fall as the detection process bases on analysis of falling images.

Limitations of Image Processing Based Fall Detection Systems

- Have limitations on pervasive detection.
- The detection area is limited by the surrounding environment.
- May become costly.
- People may have an issue with it due to the chance of privacy-compromising.

2.9.4 Smartphone Based Fall Detection System

Due to the availability of modern technology and embedding of sensors in smartphones, the number of fall detection mobile applications has increased in recent years. Based on the type and cost, a smartphone can have multiple sensors embedded in it. The available sensors are accelerometer sensor, GPS sensor, gyroscope sensor, orientation sensor and proximity sensor.

Initial smartphone based fall detection systems were developed in Symbian OS on Nokia phones [27]. In [28], Java multiplatform software architecture is employed using external architecture in both a Symbian phone (Nokia 5800) and Android smartphones (Samsung Galaxy) to detect fall. At present, Fade, Emergency fall detector are two notable smartphone-based application for detecting fall. Besides in [29] Yu and in [30] Noury has summarized the general principles of fall detection for senior citizens for android platform. But a complete review and comparison of smartphone-based fall detection and comparison is given by Habib et al. in [34].

Limitations of Prevailing Fall Detection Systems

- The smartphone's position is kept fixed to use a fixed threshold.
- Not dynamic in nature.
- Have an unbalanced co-relation between false positivity and false negativity.

2.10 Conclusion

In this project "To Design and Implement a Smartphone Based Dynamic Fall Detection System" a smartphone based fall detection application "Fall Detector" has been developed in android platform. This application defers with the prevailing application in respect of its dynamic character. The application can detect smartphone's position. And in an event of fall, it can detect fall and check user condition. Upon receiving no response from the user the application will communicate with the concerned person within short span of time. All the process is described fully in Chapter 3. In a nutshell, this application is an exertion to overcome the limitations of prevailing fall detecting applications and to provide a useful and reliable system to ensure a better life.

Chapter 3

Methodology of the Proposed System

In this work, we focused on designing a smartphone based dynamic fall detection system and implement it by developing a smartphone application by considering multiple inbuilt sensor values. This chapter mainly focuses on the overall system architecture of the proposed system and procedure to achieve this in details.

3.1 Introduction

The android application Fall Detector is a Smartphone-Based Dynamic Fall Detection System, that is developed for the purpose of ensuring better life experience. All human experience falls more or less in their life. And it results either in an instant injury or in post damnification. This system will monitor the user's present condition and check for the occurrence of fall continuously. And if a fall is detected, the app will communicate with the caregiver within a short period of time with victim's location for getting quick aid. The application will work in the background. So, the user can use his/her smartphone for his/her own task. The app will not hinder the user's normal activities.

3.2 System Overview of the Proposed System

In this work, our main concern is to design a system that can detect fall dynamically. To achieve this, that is to make the system dynamic, from the very beginning of our system multiple sensors like accelerometer, microphone and gyroscope will be initiated.

Then our first task will be to detect the phone position. With the help of accelerometer sensor's value, the smartphone's position will be detected. But there will also be a provision of training the system to get the smartphone's position according to the user's movement. And these training results will be stored and used to obtain the user specific accurate phone position.

Having phone position detected, we will set the threshold values first to check the fall. The threshold values will be set according to the smartphone's position. The likelihood of happening a fall is checked continuously for every sensor data. If the sensor's value exceeds the threshold value, then a fall is triggered by the accelerometer sensor.

Microphone sensor will also be initiated at the very beginning of the system. From that moment microphone sensor will continuously record and analyze the captured audio. If the audio signal exceeds the predefined threshold, a fall is triggered by the microphone sensor.

Similarly, orientation sensor will also be initiated at the beginning of the system. Upon starting, it will calculate the amplitude of acceleration in the total vertical direction with the help of accelerometer sensor. For every sensor data, the calculated amplitude is checked against the orientation threshold. If condition fulfills, a fall is triggered by the orientation sensor.

If any two sensor triggers at a time, the system will check for user movement for 30 seconds. If the stationary condition is found out, only then a fall is considered to have happened. And system will check for user response. Depending on the user response, the system will either go for post fall detection phase or will continue to monitor the user.

Lastly, upon deciding a fall has occurred the system will sound an alarm for one minute to get the attention of the user. If the user responds by stopping the alarm, the system will automatically go back to the monitoring phase. But if the alarm is not stopped, then the system will send SMS to the previously given care-giver's contact number with the user current location.

The abstract view of the proposed system is given below figure 3.1.

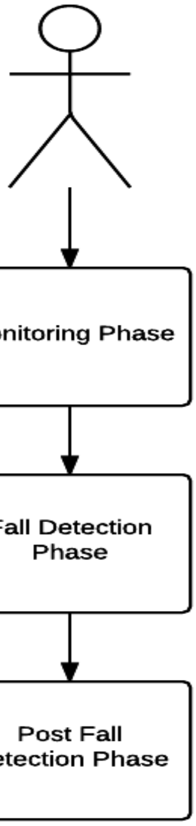

Figure 3.1: Abstract view of system architecture

In order to have a clear understanding of the proposed system, we subdivided the whole system into mainly three parts-

- Monitoring phase
- Fall detection phase
- Post fall detection phase

3.3 System Functions

In this section, the functions of the system are described in a sequencing manner. In section 3.3.1 monitoring phase is described, in section 3.3.2 fall detection phase is described and in section 3.3.3, post fall detection phase is described.

3.3.1 Monitoring Phase

In this phase, as shown in below figure 3.2, the system monitors the user's present status. This phase comprises of –

- Data collection module
- Training module

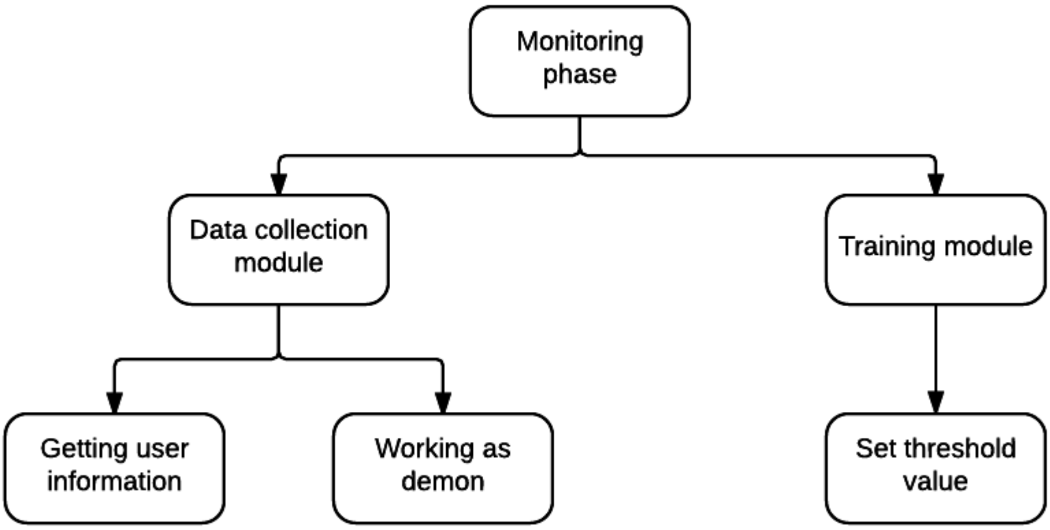

Figure 3.2: Monitoring phase of the system

Data Collection Module

In data collection module, the system collects data from multiple sensors and from the user. The phase is subdivided into two parts-

- Getting user information
- Working as module

Getting User Information

After getting started, the system will ask the user to fill up the care giver's contact numbers of his/her preference in case of future emergency communication. This contact information will be saved securely.

Working as Daemon

In this phase, the system will work in the background of the smartphone. There it will collect the sensors data. In our designed system, smartphones built-in sensor accelerometer, microphone and orientation sensor will be registered and listened from the very beginning of the monitoring. And in this phase the sensors data will be saved for future processing. In our implemented system, due to unavailability of orientation sensor in the experimenting smartphone, only accelerometer and microphone sensors are registered and listened.

3.3.1.2 Training Module

This module is dedicated to the user. The purpose of this module is to monitor the user for the given period of time for given smartphone position. Depending on the phone position the system will set the threshold value for fall triggering from the reading of the accelerometer sensor's value.

Set Threshold Value

In this phase, the obtained total acceleration values of the smartphone will be used to set the threshold value for detecting smartphone position. This threshold value will be dependent on the user's movement. As per the user's body movement, the thresholds for x-axis, y-axis and z-axis will be set respectively.

If the training module is not chosen then the threshold values will be the predefined ones. For shirt's pocket, it is <10 for accelerometer difference in x-axis, y-axis and z-axis. And for pants pocket, it is >15 for accelerometer difference in x-axis and z-axis. And >10 for accelerometer difference in y-axis.

3.3.2 Fall Detection Phase

In this phase, as shown in below figure 3.3, the obtained sensors value will be analyzed and continuously checked for fall occurrence. The phase can be subdivided into two parts-

- Data analysis module
- Sensor fusion module

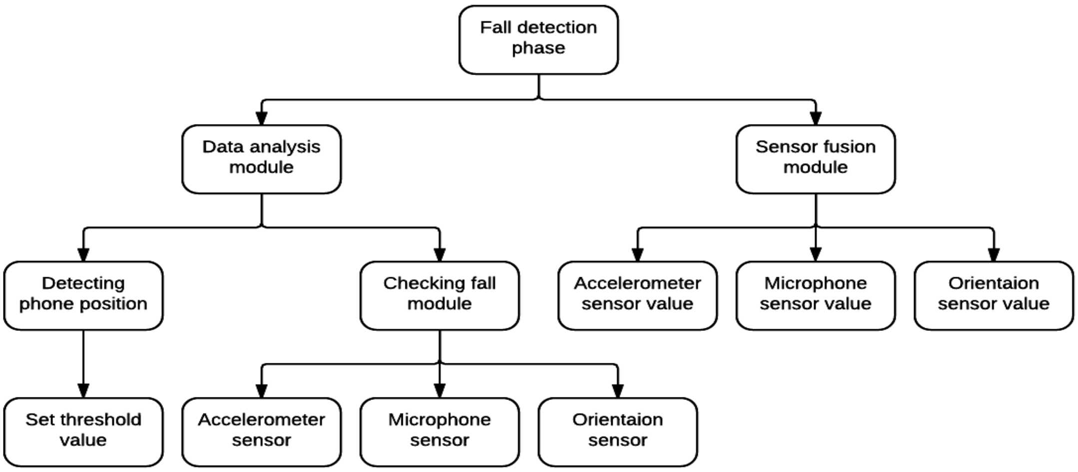

Figure 3.3: Fall detection phase of the system

3.3.2.1 Data Analysis Module

In this module, the data obtained from the accelerometer sensor, orientation sensor and microphone sensor will be analyzed. Accelerometer sensor's value will be analyzed to get the total acceleration of the user, orientation sensor's value will be analyzed to obtain the amplitude of the acceleration in the absolute vertical direction and microphone sensor value will be analyzed to get the sound level of the surrounding environment.

Detecting Phone Position

Using the accelerometer sensor's value for x-axis, y-axis and z-axis the smartphone position is detected. Starting from the initialization of the system, the system will monitor for consecutive thirty seconds and record the accelerometer sensor's value for x-axis, y-axis and z-axis. The detection is done by checking the difference of the maximum and minimum of the axis's acceleration against the predefined threshold th_{pos}. In general, the threshold values for th_{pos} are 10 for accelerometer difference in x-axis, y-axis and z-axis for shirt's pocket. And for pants pocket, it is 15 for the accelerometer difference in x axis and z axis. And 10 for accelerometer difference in y axis.

Setting Threshold Value

Based on the smartphone position, the threshold value th_{im} for fall triggering is set. At first, the total acceleration is checked against threshold th_t. If the acceleration at an instance degrades below th_t, then the system checks acceleration against threshold th_{im} for 1 second following the first threshold check. Because due to fall, the value of total acceleration at first is almost tends to zero for a perfectly free fall and for falling from a great height. Then it starts picking up. In most cases, exceeds the previous range. But for human fall on the ground from a height of 3-4 feet, the value of the total acceleration doesn't tend towards zero. But surely it goes below 1g.

Checking Fall Module

In this module, the obtained sensor values are processed to get the desired output and then undergo some conditions to check for fall.

Accelerometer Sensor

The total acceleration of the device is calculated by the equation (1).

$$|A_T| = \sqrt{(|A_X|^2 + |A_Y|^2 + |A_Z|^2)} \ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

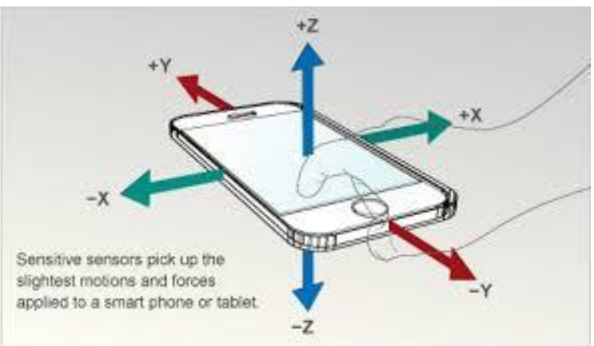

Figure 3.4: Smartphone's accelerometer sensor

An accelerometer sensor usually provides the acceleration readings in x, y and z directions like the figure 3.4. So, accelerations in these directions are termed as respectively A_X, A_Y and A_Z. And vector $|A_T|$ represents the amplitude of total acceleration of the smartphone's body. By this equation (1), the total acceleration of the user can be calculated. And the obtained acceleration can be used for further processing.

A free fall event is distinguished from human fall event by observing the accelerometer sensor's data. In case of free fall event, the magnitude of acceleration tends to zero. By identifying this character a free fall event is distinguished. In figure 3.5 and 3.6 they are pictured.

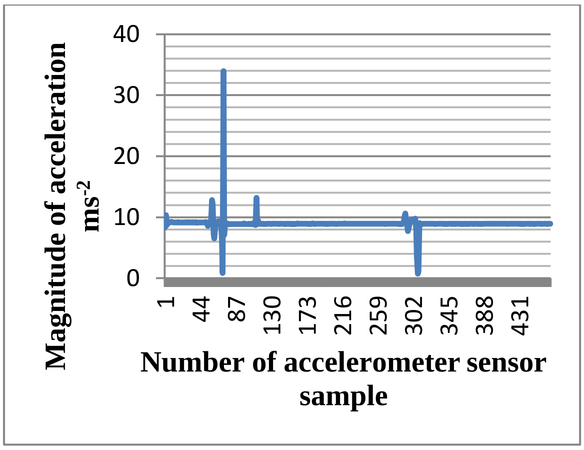

Figure 3.5: Accelerometer sensor's data due to free fall

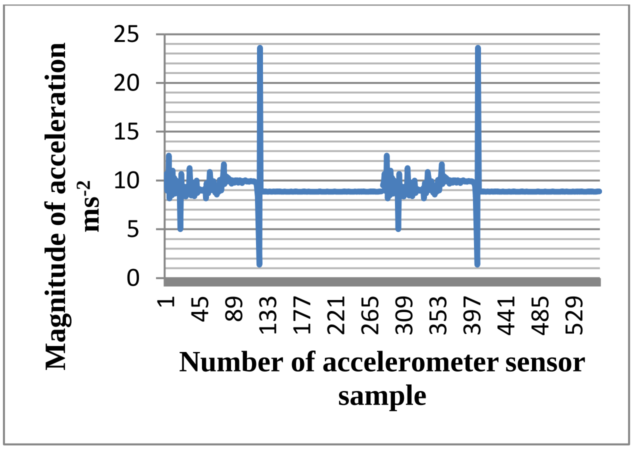

Figure 3.6: Accelerometer sensor's data due to human mimicked fall

Orientation Sensor

The inbuilt orientation sensor of the smartphone gives information about the device's position compared to a reference plane. The sensor gives output as azimuth/yaw, pitch and roll in degrees as shown in below figure 3.7. Using these values the amplitude of the acceleration in the absolute vertical direction is calculated by using the equation (2).

$$|A_V| = |A_X \sin \theta_Z + A_Y \sin \theta_Y + A_Z \cos \theta_Y \cos \theta_Z| \ldots\ldots\ldots\ldots\ldots (2)$$

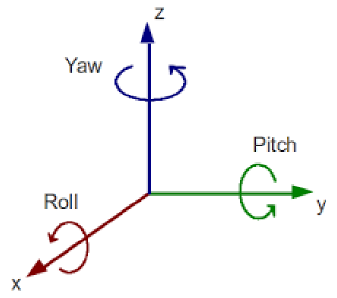

Figure 3.7: Orientation sensor value

The azimuth is the angle between the longitudinal axis of the smartphone and the earth's magnetic field. If the top of the smartphone faces north, then the value of azimuth is considered as zero degrees. So, if the top of the smartphone faces south, then the value is considered as 180 degrees.

The pitch is the angle between the smartphone's plane and the plane parallel to the ground with respect to the top or bottom of the smartphone. So, pitch basically measures the tilting of the top and the bottom of the device. And the roll is the angle that is obtained by tilting the smartphone either right or left.

Microphone Sensor

The surrounding sound level can be obtained by calculating the sound captured in the microphone sensor. The captured analog sound signal is obtained as a digital signal from the audio recorder. That is the sound signal obtained is already sampled by the built-in API (Application Programming Interface) Audio Record of the smartphone.

Then the sampled signal is saved in an array whose elements are the amplitudes of the recorded signal. Then the pressure on the microphone is calculated from the recorded amplitudes of the signal. It is considered that the amplitude is relative to the pressure experienced by the microphone. The maximum sound level that can be detected by smartphone is 90 dB. This corresponds to a pressure of 0.6325 Pa. and the minimum 0 dB corresponds to a pressure of 0.0002 Pa. So, the reference value is taken as 0.00002 Pa. And the maximum magnitude that can be saved in the array is 32,767 and the minimum magnitude is 0. So, the obtained sampled signal's amplitude value is converted into pressure by the equation (3).

$$pressure = \ amplitude/51805.5336 \ \ldots\ldots\ldots\ldots.. (3)$$

Here, the divisor 51805.5336 comes from dividing the 32,767 (maximum amplitude that could be recorded) to the 0.6325 (the maximum pressure that could be recorded). Then the calculated pressure is transformed into dB by the help of the equation (4).

$$dB = (20 * log10\left(\frac{pressure}{\mathrm{REFERENCE}}\right)) \ \ldots\ldots\ldots\ldots\ldots (4)$$

Here, the REFERENCE value is equal to 0.00002 Pa.

3.3.2.2 Sensor Fusion Module

In this module, the system checks the condition of other sensor values. If two of the sensor's values satisfy the falling condition the system goes for post fall module.

Accelerometer Sensor Value

For every value of the accelerometer sensor, at first, the values are compared with the threshold th_t. If the total acceleration of the device goes below this threshold th_t and after 1 second it equals or exceeds th_{im} a fall is triggered by accelerometer value.

Microphone Sensor Value

The microphone sensor's value is checked against the threshold th_{dB}. If the captured sound level exceeds the threshold th_{dB}, then for next 10 samples at least 30 dB difference of undermost value is searched. If such value is found, the sound level condition for fall is fulfilled and fall is triggered for microphone sensor.

Orientation Sensor Value

The value of $|A_V|$ obtained from the orientation sensor is checked against the threshold th_{av}. If the calculated value of $|A_V|$ is greater than the threshold th_{av} for a time window of 4 seconds, then the difference between maximum and minimum value of $|A_V|$ is compared with the second threshold th_{av_trig} for another time window of following 4 seconds. If the difference is greater than the second threshold, the vertical acceleration condition for fall is considered to be fulfilled and fall is triggered by orientation sensor.

If at least two sensors fulfill the fall condition, then the system checks the difference of device's present acceleration with the acceleration value equivalent of 1g for 30 seconds. If the differences found are less than the threshold th_{diff} for most cases, then the triggered event is considered as a fall and the system considers a fall is detected. If a fall is detected in this phase, the system goes for post fall detection phase. But if the triggered event is not detected as a fall the system goes back to the monitoring phase.

3.3.3 Post Fall Detection Phase

After fall detection phase if a fall is detected, the system comes to post fall detection phase. In this phase, the system tries to get the user's feedback. On account of the feedback, the system specifies its next action.

3.3.3.1 Alarm Module

Upon detecting a fall the system sounds an alarm for one minute. If the user stops the alarm, the system goes back to the monitoring phase considering the fall as a false negative. But if the user doesn't stop the alarm, the system goes to the communication module.

3.3.3.2 Communication Module

In this module, as shown in below figure 3.8, the system instantaneously communicates with the concerned caregiver by SMS (Short Message Service) which were previously given by the user. The system will notify the concerned person about the fall of the user with his/her falling location.

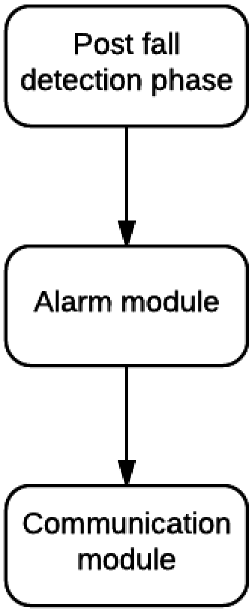

Figure 3.8: Post fall detection phase

3.4 Conclusion

In this chapter, we describe the system architecture and working procedure of the system in a great detail. Implementation of this proposed system is described in the next chapter.

Chapter 4
Implementation

The implementation of this system requires designing and development of a smartphone based software that has a Graphical User Interface (GUI) with a fall detection system on its background. In this chapter, the background tasks with necessary diagram are given which will clearly describe the outcome of this project.

4.1 Software Development

The proposed system is implemented as a mobile application. The application is developed for the android platform.

4.1.1 Development Tools

The lists of tools that have been used to implement the system are given below:

- **Software Development Kit**
    - Android
- **Integrated Development Environment (IDE)**
    - Android studio

4.2 System Layout

At first the application has to be launched by clicking the launcher icon of the application. Then a User Interface (UI) like figure 4.1 will pop up. There the user has to give the caretaker’s contact number which will be used in case of emergency.

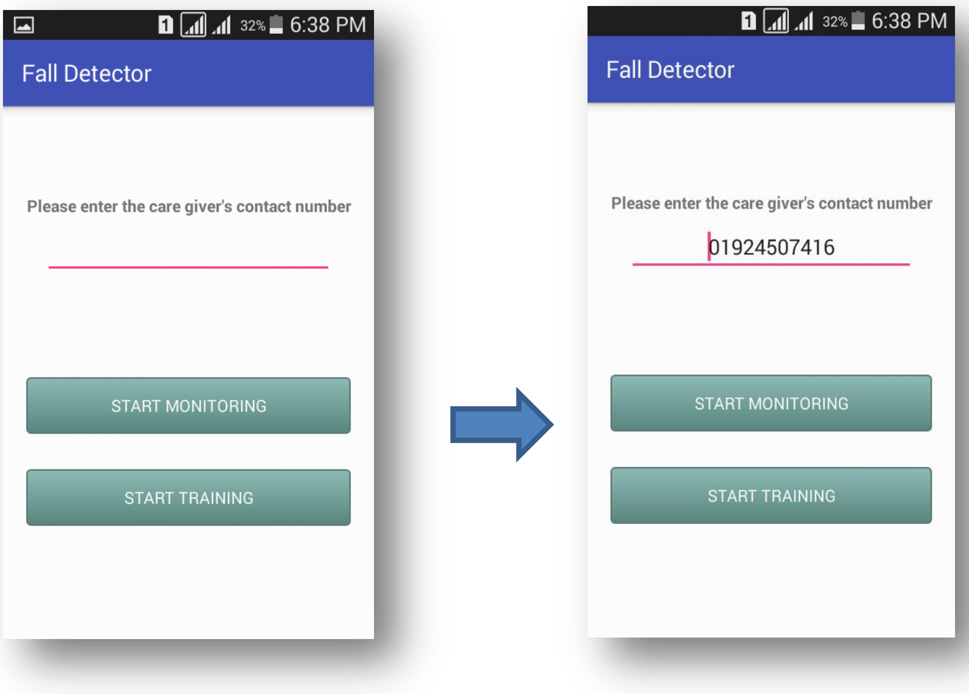

Figure 4.1: First UI

In the first UI, there will be two buttons. One for training the system and the other for monitoring the user. Upon clicking the "Start Training" button, training UI will pop up, where the user has to specify in which position and for how long the training will be conducted. The steps are pictured in below figure 4.2, figure 4.3 and figure 4.4.

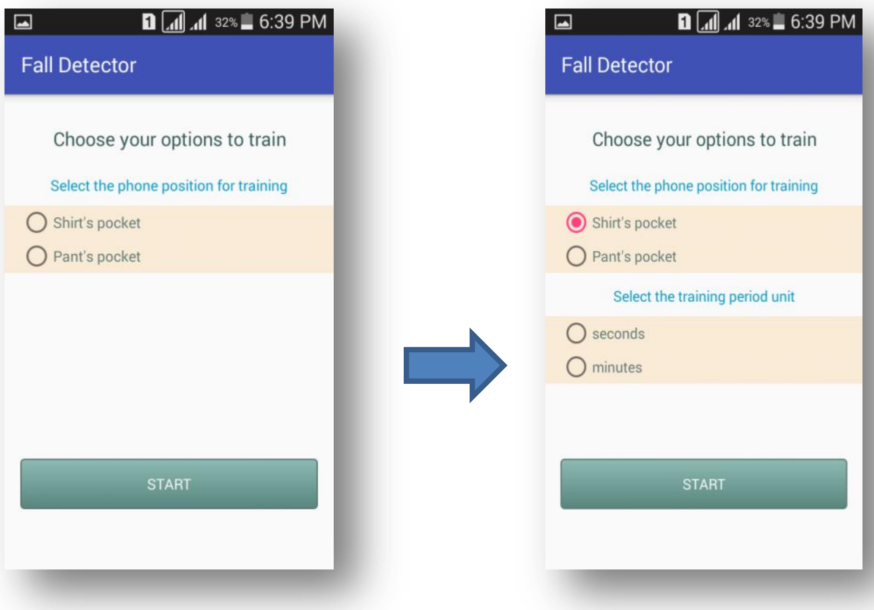

Figure 4.2: Training UI

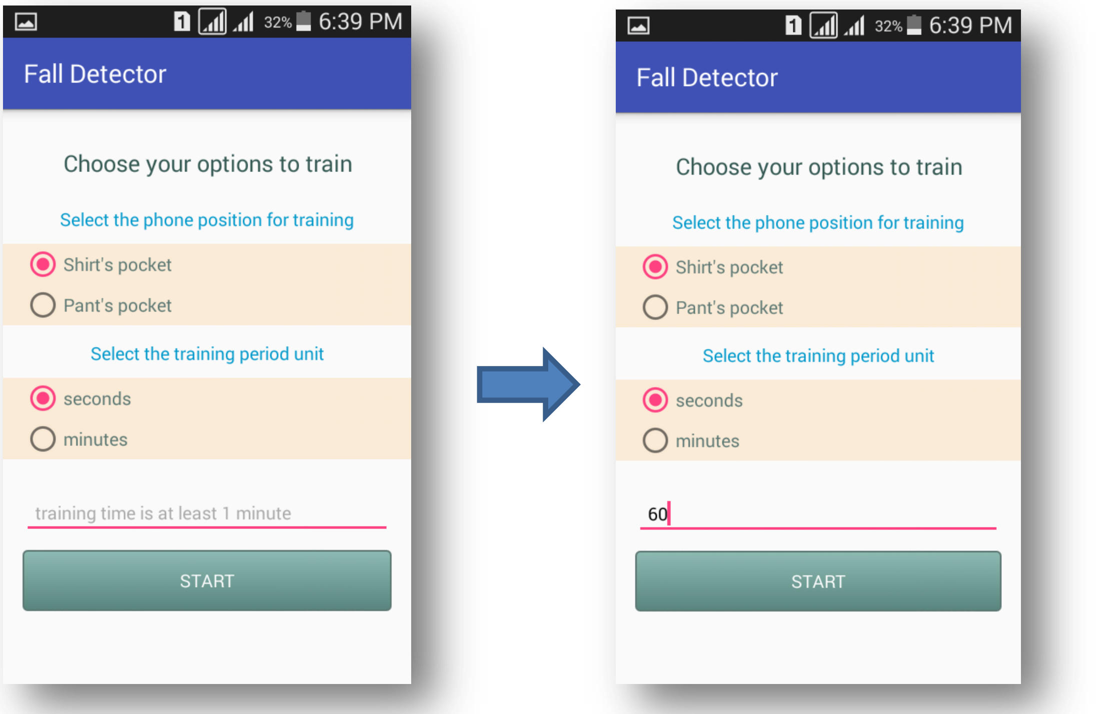

Figure 4.3: Training UI

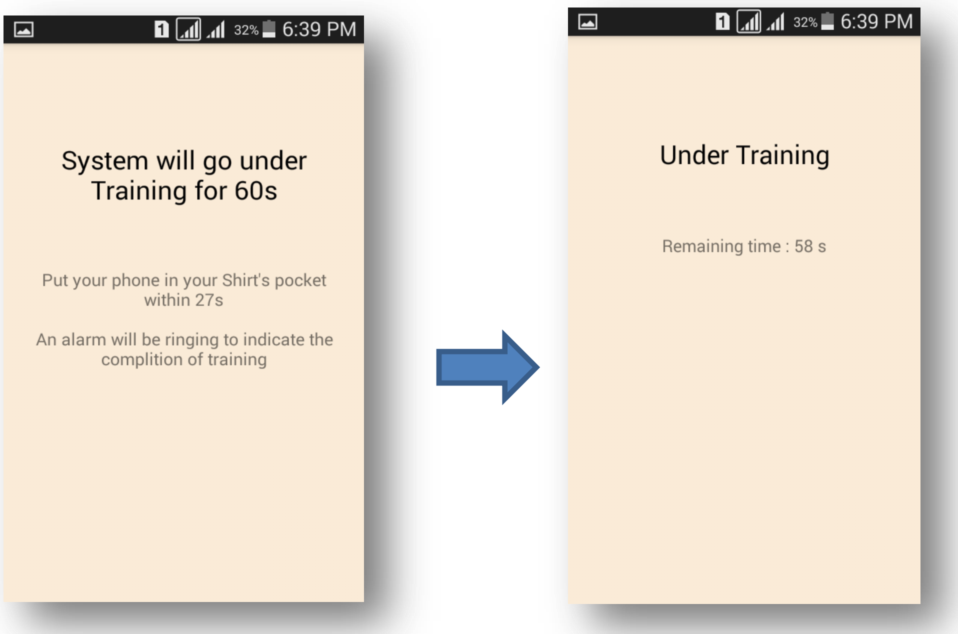

Figure 4.4: Training UI

Upon completion of the training, an alarm will be rung to notify the user about the completion of the training. A UI like figure 4.5 with related info will be popped up along with two buttons for choosing either again training or monitoring.

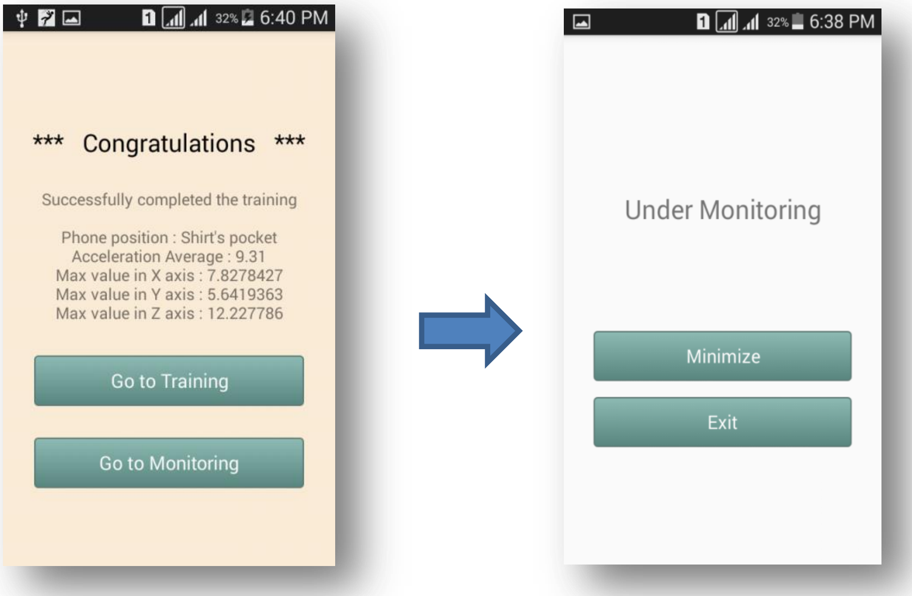

Figure 4.5: Monitoring UI

In monitoring, at first a UI like figure 4.6 will pop up to notify the user to enable the GPS if not enabled.

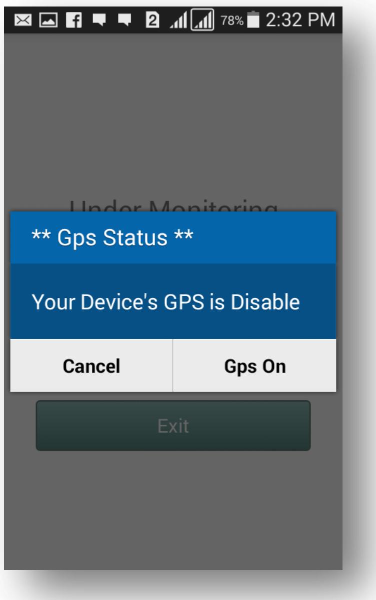

Figure 4.6: Enabling GPS UI

If fall conditions are fulfilled an alarm will be ringing and an alarm UI like figure 4.7 will be popped up to get the attention of the user.

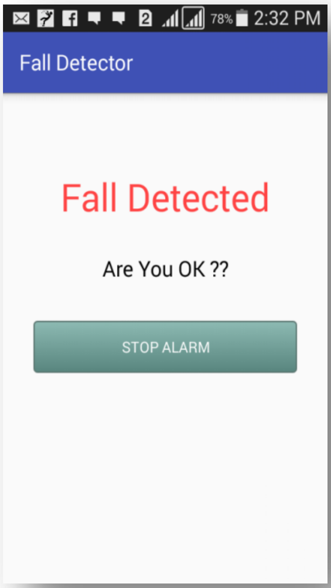

Figure 4.7: Alarm UI

If the alarm is stopped, then the system will go back to the monitoring phase. But if the alarm is not stopped, an SMS will be sent to the care giver's contact number with the user's location.

4.3 Conclusion

In this chapter, we try to give an overview of our system's implementation. Especially, the tools required to implement the system and the system's layouts.

Chapter 5

Experimental Results

We tested the system with extensive experiments. In this section, we first introduce how data are collected. Then we present the performance of the system and compare it with existing systems.

5.1 Data Collection

Data are collected from conducted experiments in two phases for two purposes. One is for setting the threshold values for detecting smartphone position. And the other is for setting threshold values for detecting human fall. For detecting smartphone position, real-life persons are used as participants. And for detecting fall, a mannequin, two human shaped dummies of different height and mattress of 6.35 cm thick are used.

5.1.1 Smartphone Position Detection Data

In this section, we visualize the data collected from the participants for detecting smartphone position. These data visualize the smartphone's accelerometer sensor's value while walking. For detecting phone position, we conducted extensive experiments on 10 different real persons of different height and weight. The participants are of 23-25 years of age. Five of them are 165-170 cm tall. Other five are 180-182 cm tall. Four of them weight within 55-65 kg, five weight 65-70 kg and one weights more than 70 kg. And the experiments are conducted for both shirt's pocket and pants pocket.

5.1.1.1 For Smartphone Placed in Shirt's Pocket

The data ranges of accelerometer in X-axis for placing the smartphone in shirt's pocket are given in below figure 5.1. The figure visualizes smartphone's built-in accelerometer sensor's value in X-axis for all 10 participants.

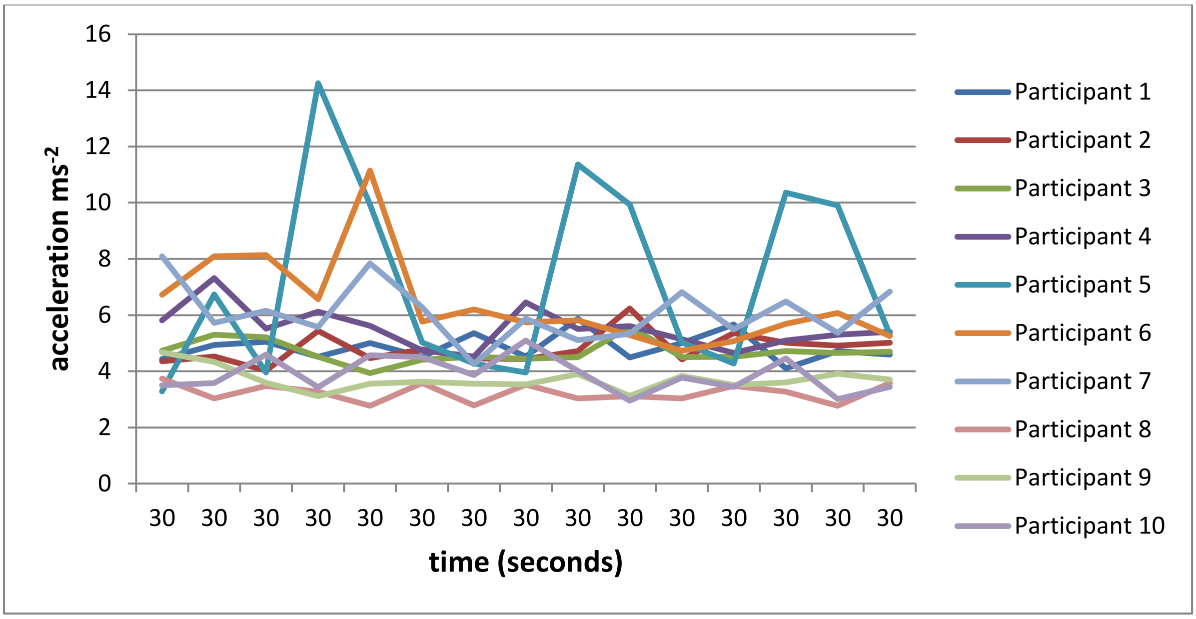

Figure 5.1: X-axis data range for smartphone placed in shirt's pocket

The data ranges of accelerometer in Y-axis for placing the smartphone in shirt's pocket are given in below figure 5.2.

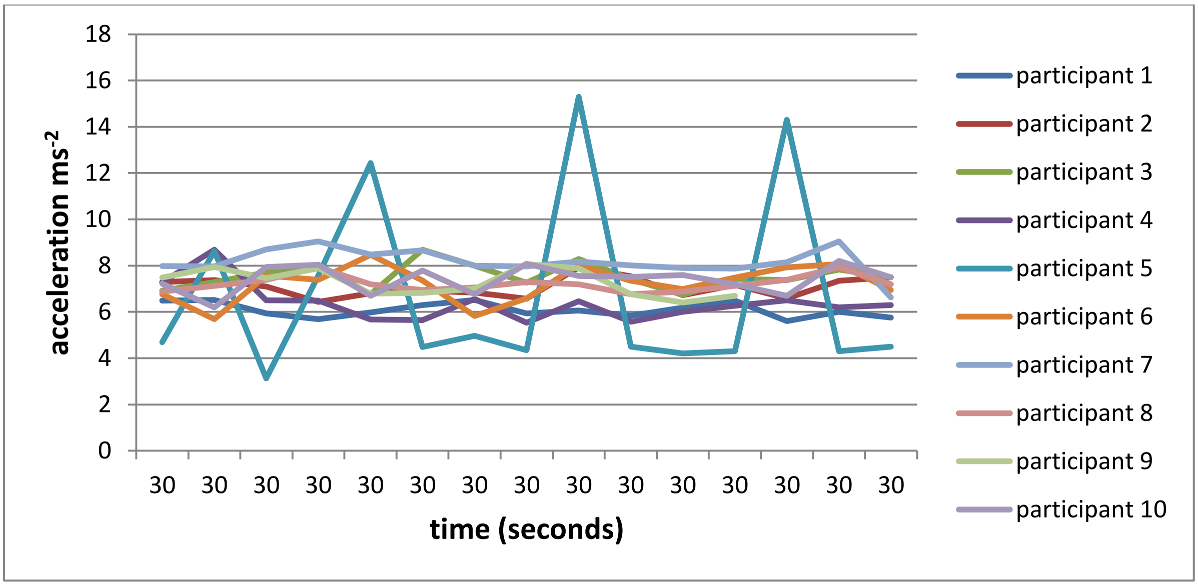

Figure 5.2: Y-axis data range for smartphone placed in shirt's pocket

The data ranges of accelerometer in Z-axis for placing the smartphone in shirt's pocket are given in below figure 5.3.

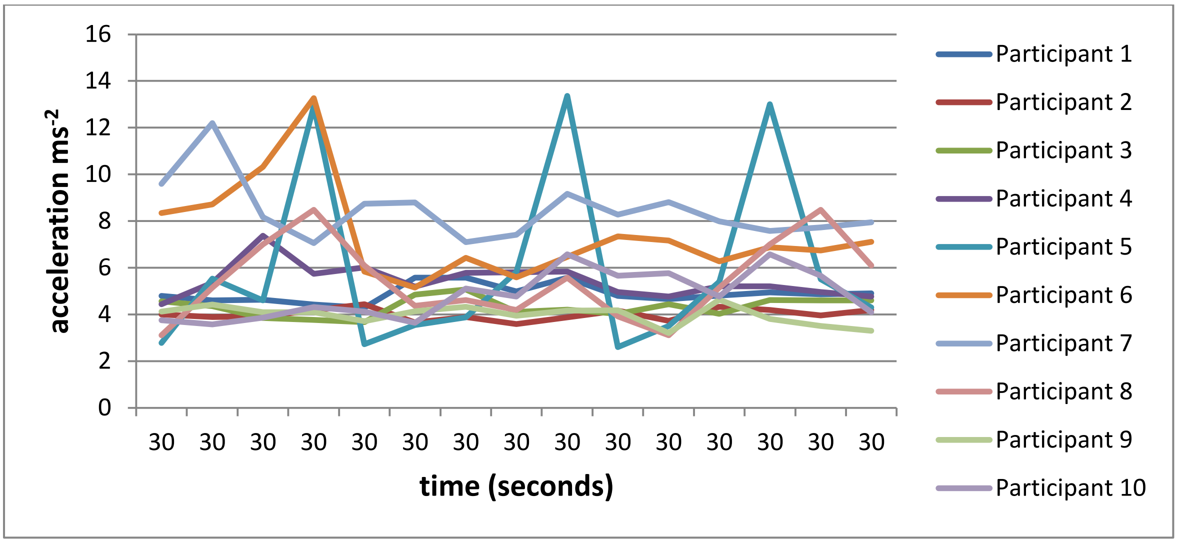

Figure 5.3: Z-axis data range for smartphone placed in shirt's pocket

5.1.1.2 For Smartphone Placed in Pants Pocket

The data ranges of accelerometer in X-axis for placing the smartphone in pants pocket are given in below figure 5.4.

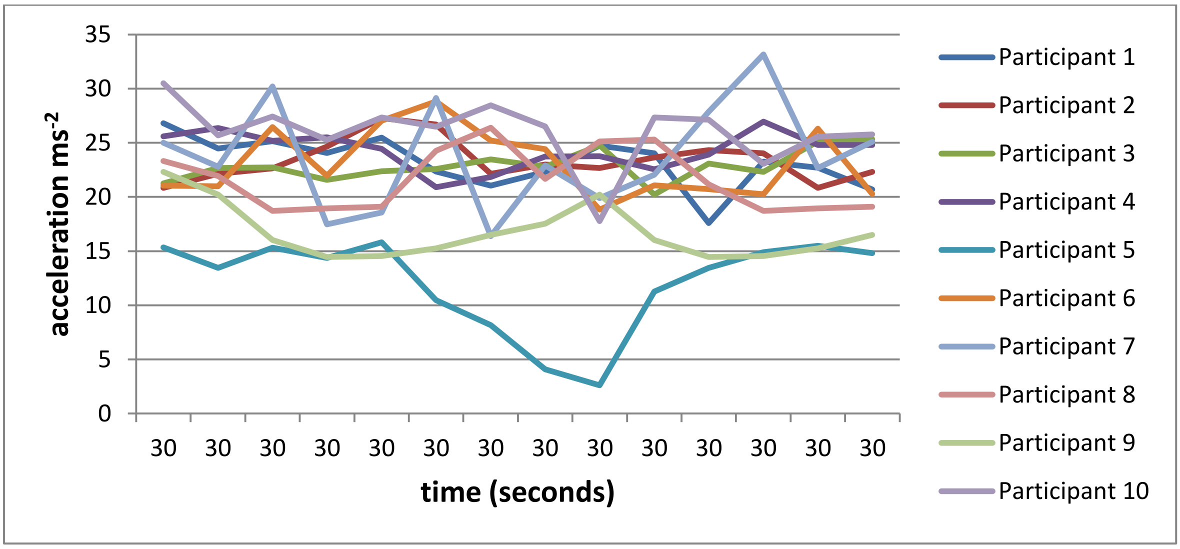

Figure 5.4: X-axis data range for smartphone placed in pants pocket

The data ranges of accelerometer in Y-axis for placing the smartphone in pants pocket are given in below figure 5.5.

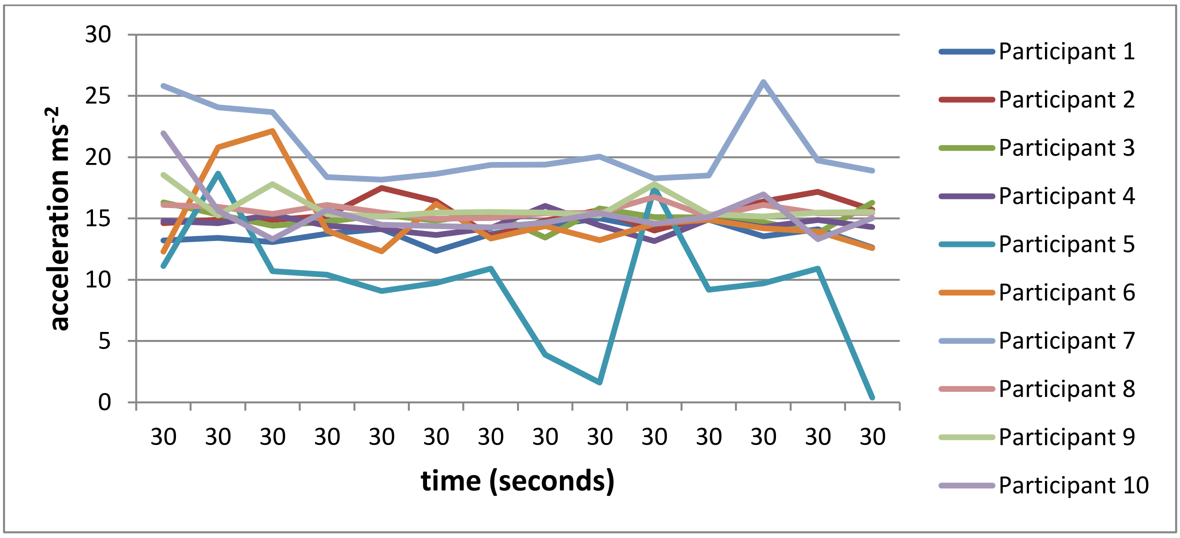

Figure 5.5: Y-axis data range for smartphone placed in pants pocket

The data ranges of accelerometer in Z-axis for placing the smartphone in pants pocket are given in below figure 5.6.

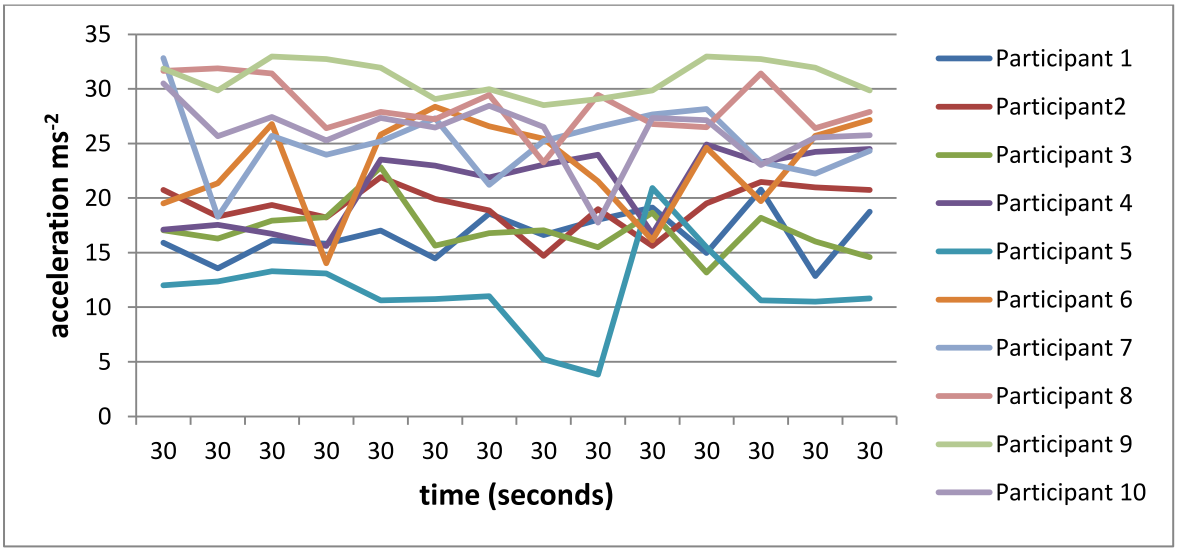

Figure 5.6: Z-axis data range for smartphone placed in pants pocket

Data obtained from participants for Smart Phone position detection are given in below table 5.1.

	Trial time (in minute)	Shirt's pocket			Pants pocket		
		X axis value range (in ms^{-2})	Y Axis value range (in ms^{-2})	Z axis value range (in ms^{-2})	X axis value range (in ms^{-2})	Y Axis value range (in ms^{-2})	Z axis value range (in ms^{-2})
Participant 1	10	4-6	5-7	4-6	20-27	12-15	13-21
Participant 2	10	4-7	6-8	3-5	20-28	13-18	14-22
Participant 3	10	4-6	6-9	3-6	20-26	13-17	13-23
Participant 4	10	4-8	5-9	4-8	20-27	13-17	15-25
Participant 5	10	3-15	4-16	2-14	2-16	1-19	3-21
Participant 6	10	4-11	5-9	5-14	18-29	12-23	14-29
Participant 7	10	4-9	6-10	7-13	16-34	18-27	18-33
Participant 8	10	2-4	6-8	3-9	18-27	15-17	26-32
Participant 9	10	3-5	6-9	3-5	14-23	15-19	29-33
Participant 10	10	2-5	6-9	3-7	17-31	13-22	17-32

Table no 5.1: Axis data range of accelerometer for 10 participants

5.1.2 Mimicking Human Fall Data

Though real-life experiment with real persons was not possible, but the system is experimented using a mannequin and two human-shaped dummy of different height. Testing objects are used to mimic the human fall.

Pictorial views of values of acceleration and sound level before, during and after mimicking human fall are given in below figure 5.7 and figure 5.8. The areas covered by orange colored dotted rectangle specify respectively the change in accelerometer sensor's and microphone sensor's value due to mimicking human fall.

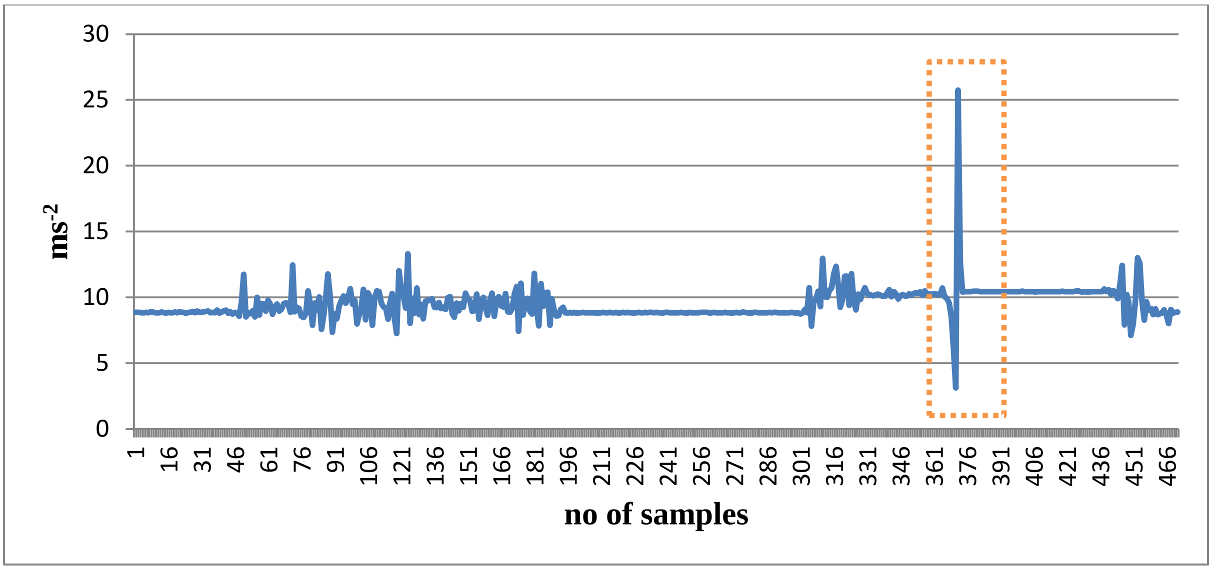

Figure 5.7: Accelerometer sensor's value before, during and after fall

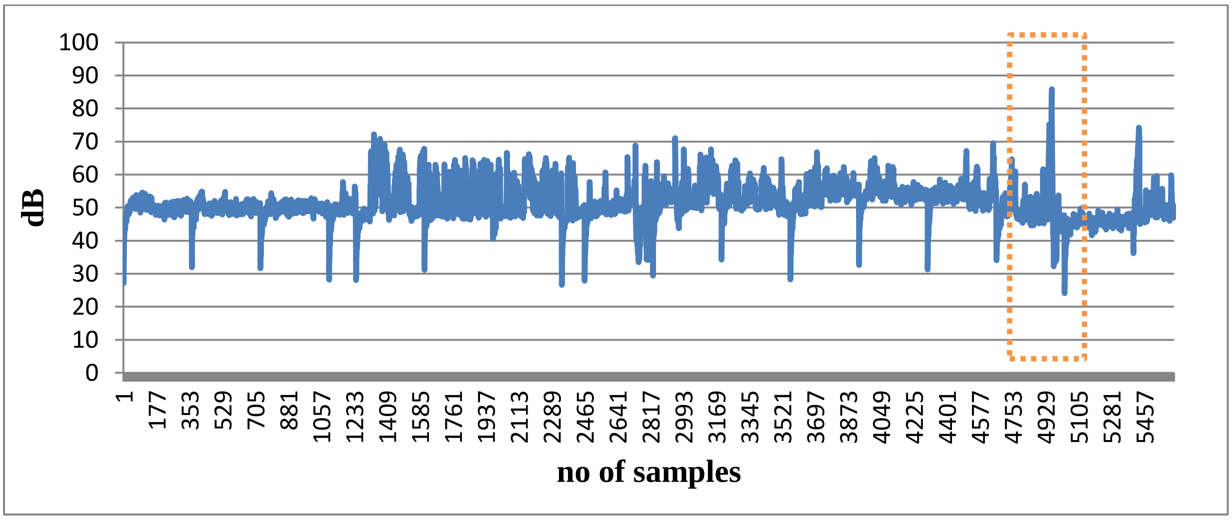

Figure 5.8: Microphone sensor's value before, during and after fall

5.1.2.1 Mannequin Test

The fall data for mannequin are obtained using a mannequin of 182 cm tall. In this experiment, the smartphone sensors are kept interdependent of each other. The experiments are conducted keeping the smartphone in shirt's pocket and pants pocket of the mannequin. The mannequin used is shown in below figure 5.9 and the obtained experimental data for both smartphone placed in shirt's pocket and pants pocket are shown in figure 5.10 and figure 5.11. Here, data for 10 trials are pictured in graphical view.

Figure 5.9: Used mannequin to mimic human fall

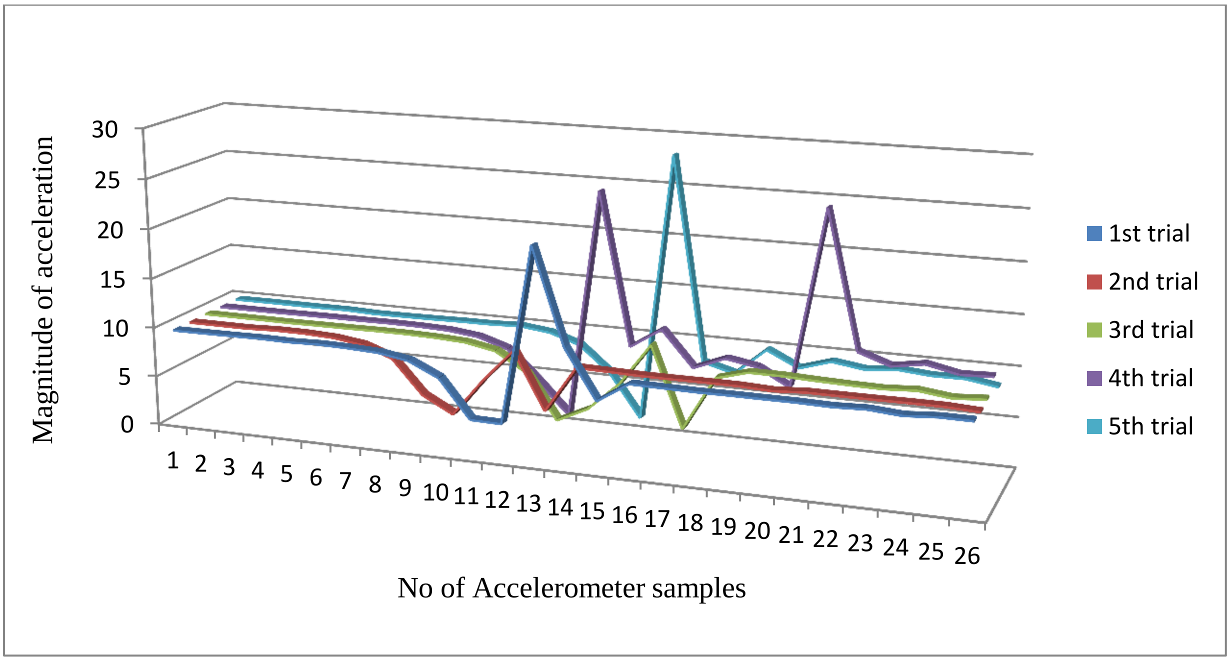

Figure 5.10: Accelerometer sensor's data for smartphone placed in shirt's pocket of mannequin

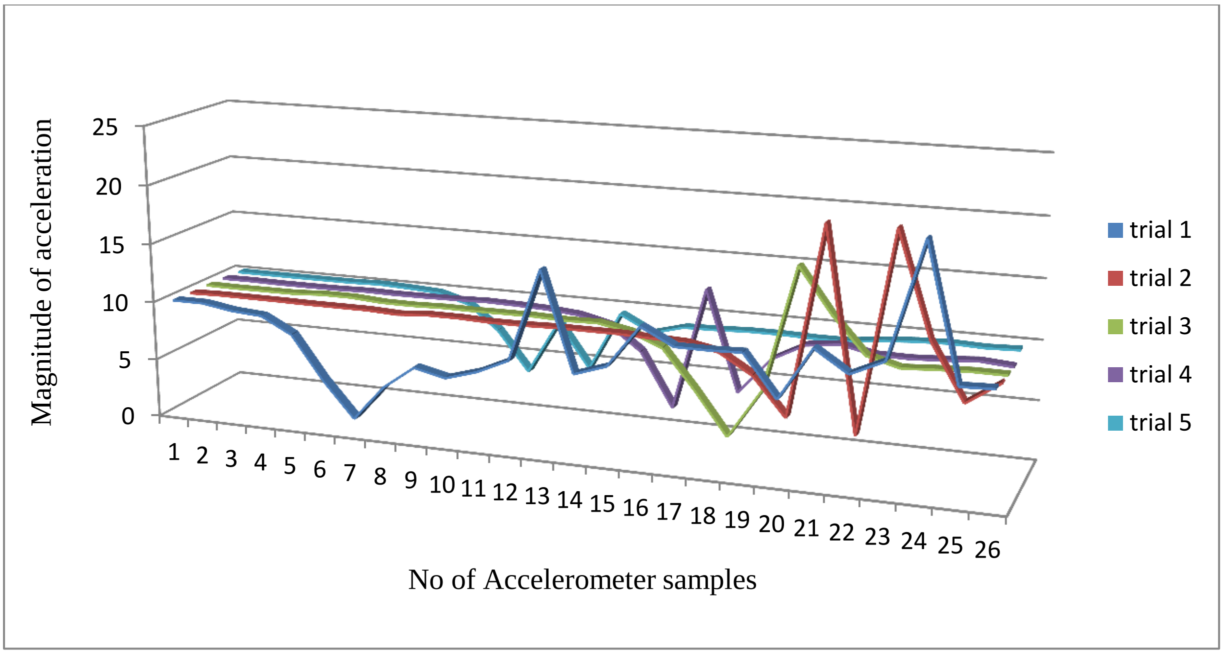

Figure 5.11: Accelerometer sensor's data for smartphone placed in pants pocket of mannequin

5.1.2.2 Human Shaped Dummy

The fall data for scarecrow shaped human dummy are obtained by using two dummy of length 155 cm and 176 cm. Here, the sensors are independent of each other. The experiments are conducted keeping the smartphone in shirt's pocket and pants pocket of the dummies. The dummy used in experiment is shown in below figure 5.12.

Figure 5.12: Human shaped dummy

Sensor data for first human-shaped dummy are given below in tabular form in table 5.2 and table 5.3.

For Smartphone Placed in Shirt's Pocket

Test No	Triggered Accelerometer value range(ms^{-2})	Triggered Microphone value (dB)
01	9.8-4.24-1.57-0.76-8.8	80.8583 - 49.1261
02	9.8-5.5-1.38-23.6-8.8	82.6055 - 47.6501
03	10.2-3.76-4.78-8.9	80.1791 - 47.9251
04	9.5-5.42-1.15-13.95-8.8	86.2160 - 51.4892
05	9.8-3.3-21.76-8.8	85.3391 - 53.3218
06	11-4.3-20.42-7.3-3.94-1.65-10.26-8.8	85.1860 - 48.7985
07	10.5-5.37-0.37-3.1-8.8	81.1150 - 46.9491
08	9.2-5.5-0.83-23.26-8.8	85.9021 - 49.9327
09	9.5-5.6-2.9-10.3	84.7467 - 50.5228
10	10.6-1.81-2.87-6.72-14.1-10.3	81.6661 - 45.6302

Table no 5.2: Sensor data of first dummy for shirt's pocket

For Smartphone Placed in Pants Pocket

Test No	Triggered Accelerometer value range(ms^{-2})	Triggered Microphone value (dB)
01	10-6.8-4.3-14.15-8.8	80.3023 - 49.6289
02	9.4-3.9-5.5-8.9	79.7551 - 47.7116
03	10-4.3-30.68-8.8	81.4088 - 49.9619
04	9.8-5.8-4.8-11.92-8.8	80.3526 - 48.0912
05	10.2-6.4-2.5-9.3-8.8	79.1303 - 42.0774
06	10-5.2-21.38-3.2-9-8.9	79.4037 - 48.0254
07	9.9-5.2-7.1-8.8	80.4588 - 38.7683
08	8.8-6.8-5.4-1.7-8.8	79.0109 - 26.6437
09	13-4.1-9-7-8.8	72.3602 - 31.3955
10	11.5-5.5-7.5-8.8	81.2497 - 49.1070

Table no 5.3: Sensor data of first dummy for pants pocket

Sensor data for second human-shaped dummy are given below in tabular form in table 5.4 and table 5.5.

For Smartphone Placed in Shirt's pocket

Test No	Triggered Accelerometer value range(ms^{-2})	Triggered Microphone value (dB)
01	9.8-7.4-4.3-1.4-2-8.8	82.7626 - 49.4779
02	9.9-7.6-4-1.2-11.8-8.9	81.6343 - 48.2639
03	10-7-4-6.9-11.3-10.3	84.3358 - 49.4224
04	9.6-7.7-4.5-1.74-1.68-9.4-8.8	85.6883 - 46.07905
05	9.8-5.7-1.6-6.7-4.6-8.8	78.1974 - 46.1443
06	19.9-5.6-2.6-6.8-10.2	79.3936 - 46.7854
07	11.4-9.3-25-3.5-5.6-10.2	79.5362 - 46.7551
08	11.3-5.4-2.2-6.2-10.4-8.8	79.0868 - 48.1032
09	12-7.5-2.6-3.3-33.9-9.7-8.8	82.7787 - 48.3302
10	10-2.3-5.2-21-8.8	72.8169 - 37.1957

Table no 5.4: Sensor data of second dummy for shirt's pocket

For Smartphone Placed in Pants Pocket

Test No	Triggered Accelerometer value range(ms^{-2})	Triggered Microphone value (dB)
01	10.2-6-5.41-8.8	-
02	10-7.9-5.6-9.2-10.3	78.0054 - 46.9752
03	10-6.2-4.9-9-8.8	76.6901 - 45.1428
04	10-6.9-4.7-32.8-8.8	77.8208 - 47.2657
05	10-7.1-4.8-28-8.9	79.3187 - 48.5401
06	11-8.2-1.6-2.1-8.8	78.5249 - 47.4757
07	10.7-6.7-17.2-6.6-10.3	74.8216 - 43.4051
08	11-7.7-5.8-5.7-1.6-8.8	80.7450 - 50.3683
09	11-8.8-14.5-7.5-4.88-7.6-18.8-8.8	76.9961 - 45.4232
10	11.2-1.7-6.3-33.9-8.8	78.2517 - 48.0408

Table no 5.5: Sensor data of the second dummy for pants pocket

The first dummy has also experimented in the normal and noisy environment. The smartphone is kept both in shirt's pocket and pants pocket. The obtained sensor data after mimicking human fall are shown in below tables table 5.6, table 5.7, table 5.8 and table 5.9.

For Smartphone Placed at Noisy Environment

Pant pocket	**Acceleration (ms^{-2})**	**Sound level (dB)**	**dB difference**
	10-4-10	55-69-43-51	26
	10-4-10	53-69-38-52	31
	10-3-20-3-9	53-67-41-50	26
	10-5-11-8-10	52-82-35-52	47
	10-5-13-9	54-72-42-50	30
	10-5-21-11	52-72-38-52	34
	10-5-21-11	53-65-46-52	19

Table no 5.6: Sensor data of first dummy for pants pocket in noisy environment

Shirt pocket	**Acceleration (ms^{-2})**	**Sound level (dB)**	**dB difference**
	10-3-23-10	50-70-32-50	38
	10-3-14-9	50-74-32-50	42
	10-3-33-9	45-69-31-41	38
	10-1-23-9	55-78-33-51	45
	10-2-32-9	55-71-31-51	40
	10-2-9	55-84-32-55	52
	10-3-31-10	54-65-31-52	34
	10-4-15-10	55-70-31-51	39
	10-3-33-10	55-62-38-60-55	24
	10-2-10	54-45-81-33-51	48

Table no 5.7: Sensor data of first dummy for shirt's pocket in noisy environment

For Smartphone Placed at Normal Environment

Sensor's Data at shirt's Pocket

Pant pocket	Acceleration (ms^{-2})	Sound level (dB)	dB difference
	3-13-6-27-9	50-75-52-70-35-48	35
	10-2-9	50-70-34-48	34
	10-3-15-9	51-74-41-50	33
	10-2-13-9	51-78-34-48	44
	10-3-22-9	50-75-35-42	40
	10-2-12-9	48-78-28-45	50
	10-4-40-9	49-70-33-45	37

Table no 5.8: Sensor data of first dummy for pants pocket in normal environment

Sensor's Data at Pants Pocket

Shirt pocket	Acceleration (ms^{-2})	Sound level (dB)	dB difference
	10-3-34-9	35-67-21-40-25	37
	10-2-9	34-68-19-40-27	49
	10-3-10	35-68-28-32	40
	10-3-24-9	30-66-32	34
	10-2-10-1-9	53-78-32-50	46
	10-2-27-9	50-70-30-47	40
	10-3-9	50-71-31-48	40
	10-2-9	50-80-35-50	45
	10-2-12-9	52-65-28-45	37

Table no 5.9: Sensor data of first dummy for shirt's pocket in normal environment

5.2 Evolution of the System

The obtained sensor data are used to evaluate the system. The system is evaluated on the basis of two key points.

1. Sensitivity
2. Specificity

Sensitivity refers to true positive rate. It is also known as the probability of detection. That is sensitivity measures the proportion of positives that are correctly identified. And specificity refers to true negative rate. It measures the proportion of negatives that are correctly identified. The equation (5) is used to calculate sensitivity and equation (6) is used to calculate specificity.

$$Sensitivity = \frac{Number\ of\ True\ Positive}{Number\ of\ True\ Positive + Number\ of\ False\ Negative} \ldots\ldots\ldots\ldots\ldots. (5)$$

$$Specificity = \frac{Number\ of\ True\ Negative}{Number\ of\ True\ Negative + Number\ of\ False\ Positive} \ldots\ldots\ldots\ldots\ldots.. (6)$$

In general, positive means identified and negative means rejected. And there are four possible outcomes of positivity and negativity. They are

1. True positive (correctly identified)
2. False positive (incorrectly identified)
3. True negative (correctly rejected)
4. False negative (incorrectly rejected)

Here, sensitivity is used to measure the proportion of human falls that can be identified correctly. And specificity is used to measure the proportion of free fall that is identified correctly. So, true positive means human fall identified as human fall, false positive means free fall identified as human fall, true negative means free fall identified as free fall and false negative means human fall identified as free fall.

5.2.1 Experimental Results of the System

The data obtained from experimenting the system are shown in the below table 5.10.

	Smartphone position	Number of trials	Number of fall detected	Number of false positive	Accuracy (%)	Error rate (%)
Dummy no 01	Shirt's pocket	24	20	04	83.33%	16.66%
	Pants pocket	29	28	01	96.55%	3.44%
Dummy no 02	Shirt's pocket	10	19	01	90%	10%
	Pants pocket	10	18	02	80%	20%

Table no 5.10: Experimental results of the system

5.2.2 Overall Performance of the System

The overall performance of the system can be determined by calculating the sensitivity and specificity of the system. The sensitivity of the system can be calculated by using equation (5) and specificity by using equation (6).

The sensitivity of the system = $\frac{Number\ of\ True\ Positive}{Number\ of\ True\ Positive + Number\ of\ False\ Negative}$

$= \frac{65}{65+8}$

= 89.04%

The specificity of the system = $\frac{Number\ of\ True\ Negative}{Number\ of\ True\ Negative + Number\ of\ False\ Positive}$

$= \frac{15}{15+5}$

= 75%

	Number of trials	Estimated result	Deviation	Sensitivity	Specificity
Free fall	20	15	5	89.04%	75%
Mimic of human fall	73	65	8		

Table no 5.11: Overall performance of the system

5.3 Comparison with Existing Systems

There are some similar systems available in Google Play store. A comparison is made between two available systems and the implemented system in the below table 5.12.

	Fade	Emergency fall detector	PerFallD	Fall Detector (implemented system)
Multiple sensor	×	×	✓	✓
Accelerometer used to detect fall	✓	✓	✓	✓
Microphone used to detect fall	×	×	×	✓
Continuous monitoring	×	✓	×	✓
False fall tolerant	×	×	✓	✓
Works on locked state	×	✓	-	✓
Sensitivity	100%	100%	-	89.04%
Specificity	0%	0%	-	75%

Table 5.12: Comparison with existing systems

5.4 Discussion

The primary focus of this work is to detect human fall irrespective of smartphone position and reduce the post-fall affects by placing almost an instant communication. After numerous testing, it is marked that, for accelerometer sensor, values between 0.1g and 0.56g give the optimal result. And for microphone sensor, values greater than 60dB having at least 30 dB difference of undermost value give the optimal result. So, these values are taken as threshold values respectively for accelerometer and microphone sensor. Testing with these values gives the sensitivity of 89.04% and specificity of 75%.

Chapter 6

Conclusion and Future Recommendation

In this chapter in section 6.1, we conclude our developed system. We describe the future recommendations for further improvements of our developed system in section 6.2.

6.1 Conclusion

Our primary aim was to develop a hassle-free robust system that can detect human fall. We build a system which can detect human fall requiring no wearable sensors or hardware rather using only the smartphone and based on some important physical parameters. The system also monitors the human behavior as a pre-requisite of the fall detection. Our effort is to let the system workable within real environments. We relied on multiple sensors data and used simpler ways to monitor human behavior. This definitely contributes to speed up the overall system performance. For ensuring safety and reducing the post-fall damnification in human life, it is important to detect fall promptly and get medical aid in time. If the system can detect fall correctly and place communication on the basis of human feedback then it will be a shield to reduce the high injury death rate and injury-related hospitalization rate due to fall.

6.2 Future Recommendations

We have tried our level best to differentiate between normal free fall and human fall and to detect fall only for human fall. But it is not an easy task. The system works fine for most of the cases. But, for some cases of free fall with less acceleration but not sufficiently less, the system gives inaccurate result. We have tested the system by mimicking human fall, not observing actual human fall. So, for real life human fall detection the thresholds may need to be updated. So, developing a smartphone power friendly system is of future task.

This work is in its inceptive stages, and there are still some promising dimensions to explore. To let the system work with more intelligence, different approaches to add more sensors as parameter can be taken. Height change, location change can be of such interesting parameters in case of detecting human fall. Adding of more sensors can increase the efficiency of the system. The future recommendations are

- Setting thresholds observing real human fall
- Distinguishing human fall from free fall more accurately
- Considering more fall detector parameters like height change, location change
- Working with more in-built sensors

If the system is implemented in different types of smartphone operating system then hopefully the high fatality rate of human fall will be reduced in a salient way.

References

[1] World Health Organization (WHO)'s official site [Online], Available at http://www.who.int/mediacentre/factsheets/fs344/en/ Accessed 8 November 2017.

[2] Alexander BH, Rivara FP, Wolf ME, "The cost and frequency of hospitalization for fall–related injuries in older adults", in proc. American Journal of Public Health, 1992; 82(7): pp.1020–3.

[3] Sterling DA, O'Connor JA, Bonadies J. "Geriatric falls: injury severity is high and disproportionate to mechanism", in proc. Journal of Trauma–Injury, Infection and Critical Care, 2001; vol 50(1): pp.116–9.

[4] Hayes WC, Myers ER, Morris JN, Gerhart TN, Yett HS, Lipsitz LA. "Impact near the hip dominates fracture risk in elderly nursing home residents who fall", in proc. Calcified Tissue International, 1993;vol 52: pp.192-198.

[5] Jager TE, Weiss HB, Coben JH, Pepe PE. "Traumatic brain injuries evaluated in U.S. emergency departments, 1992–1994", in proc. Academic Emergency Medicine 2000&359; vol 7(2): pp.134–40.

[6] World report on child injury prevention, Geneva: World Health Organization (WHO), ISBN 978 92 4 156357 4, 2008.

[7] Bangladesh Health and Injury Survey Report on Children, Directorate General of Health Services, Ministry of Health and Family Welfare, Government of the People's Republic of Bangladesh, Published: January 2005, ISBN 92-806-3855-6.

[8] Centers for Disease Control and Prevention, National Center for Injury Prevention and Control[Online], Available at https://www.cdc.gov/homeandrecreationalsafety/falls/adult falls.html Accessed 15 November, 2017.

[9] Burns EB, Stevens JA, Lee RL. "The direct costs of fatal and non-fatal falls among older adults—United States", in proc. Journal of Safety Research, 2016:58.

[10] Vellas BJ, Wayne SJ, Romero LJ, Baumgartner RN, Garry PJ. "Fear of falling and restriction of mobility in elderly fallers", Age and Ageing 1997; vol 26: pp.189–193.

[11] Nyan MN, Tay FE, Manimaran M, Seah KH. "Garmentbased detection of falls and activities of daily living using 3-axis MEMS accelerometer", in proc. Journal of Physics Conference Series , 2006,vol 34: pp.1059–1067

[12] Kangas M, Konttila A, Lindgren P, Winblad I, Jamsa T. "Comparison of low-complexity fall detection algorithms for body attached accelerometers",in proc. Gait Posture , 2008,vol 28(2): pp.285–291

[13] Brown G. "An accelerometer based fall detector: development, experimentation, and analysis", SUPERB internship report, University of California, Berkeley, 2005.

[14] Hwang JY, Kang JM, Jang YW, Kim HC. "Development of novel algorithm and real-time monitoring ambulatory system using Bluetooth module for fall detection in the elderly.", In: Proceedings of 26th annual international conference of the engineering in medicine and biology society, IEEE-EMBS 2004.

[15] Pannurat N, Thiemjarus S, Nantajeewarawat E. "Automatic Fall Monitoring: A Review. *Sensors",* 2014, *14*, pp. 12900–12936.

[16] Yuan B, Herbert J. "Non-Intrusive Movement Detection in Cara Pervasive Healthcare Application", In Proceedings of the 2011 International Conference on Wireless Networks, (WORLDCOMP), Las Vegas, NV, USA, 18–21 July 2011; pp. 360–366.

[17] Honglun H, Meimei H, Minghui W. "Sensor-Based Wireless Wearable Systems for Healthcare and Falls Monitoring", *Int. J. Smart Sens. Intell. Syst.* 2013, *6*, pp.2200–2216.

[18] Shuo C. "Fall Detection System Using Arduino Fio". In Proceedings of the IRC Conference on Science, Engineering and Technology, Singapore, 13 May 2015.

[19] Available at http://www.brickhousealert.com/personal-emergency-medicalalarm.htm [Online] Accessed 8 January, 2017.

[20] Available at https://www.medicalguardian.com/product/home-guardian [Online] Accessed 15 November, 2017.

[21] Ganti RK, Jayachandran P, Abdelzaher TF, Stankovic JA. "SATIRE: a software architecture for smart AtTIRE", In: Proceedings of the 4th international conference on mobile systems, applications and services, MobiSys, 2006.

[22] Karantonis DM, Narayanan MR, Mathie M, Lovell NH, Celler BG. "Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring", in proc. IEEE Trans Inf Technol Biomed, 2006, 10(1): pp.156–167

[23] Fu Z, Culurciello E, Lichtsteiner P, Delbruck T. "Fall detection using an address-event temporal contrast vision sensor":, In: Proceedings of IEEE international symposium on circuits and systems, ISCAS, 2008.

[24] Sixsmith A, Johnson N. "A smart sensor to detect the falls of the elderly", in proc. Pervasive Comput, 2004, vol 3(2): pp.42–47.

[25] Miaou S, Sung P, Huang C. "A customized human fall detection system using omni-camera images and personal information", In: Proceedings of 1st distributed diagnosis and home healthcare conference, 2006.

[26] Jansen B, Deklerck R. "Context aware inactivity recognition for visual fall detection", In: Proceedings of pervasive health conference and workshops, 2006.

[27] Zhao Z, Chen Y, Wang S, Chen Z. "FallAlarm: Smart Phone Based Fall Detecting and Positioning System", In Proceedings of the 9th International Conference on Mobile Web Information Systems (MobiWIS), Niagara Falls, ON, Canada, 27–29 August 2012; pp. 617–624.

[28] Laguna MA, Finat J. "Remote Monitoring and Fall Detection: Multiplatform Java Based Mobile Applications", In Proceedings of the III International Workshop of Ambient Assisted (Living IWAAL 2011), Torremolinos-Málaga, Spain, 8–10 June 2011; pp. 1–8.

[29] Yu X. "Approaches and Principles of Fall Detection for Elderly and Patient", In Proceedings of the 10th International Conference on E-Health Networking, Applications and Services (HealthCom 2008), Singapore, 7–9 July 2008; pp. 42–47.

[30] Noury N, Fleury A, Rumeau P, Bourke A, Laighin G, Rialle V, Lundy J. "Fall Detection-Principles and Methods", In Proceedings of the 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS 2007), Lyon, France, 22–26 August 2007; pp. 1663–1666.

[31] Hegde R, Sudarshan B, Kumar S, Hariprasad S, Satyanarayana B. "Technical Advances in Fall Detection System—A Review. *Int. J. Comput. Sci. Mob. Comput.* 2013, *2*, 152–160.

[32] Noury N, Rumeau P, Bourke A, ÓLaighin G, Lundy J. "A Proposal for the Classification and Evaluation of Fall Detectors", *IRBM* 2008, *29*, pp.340–349.

[33] Hijaz F, Afzal N, Ahmad T, Hasan O. "Survey of Fall Detection and Daily Activity Monitoring Techniques", In Proceedings of the 2010 International Conference on Information and Emerging Technologies (ICIET), Pakistan, 14–16 June 2010; pp. 1–6.

[34] Habib MA, Mohktar MS, Kamaruzzaman SB, Lim KS, Pin TM, Ibrahim F. "Smartphone-Based Solutions for Fall Detection and Prevention: Challenges and Open Issues', *Sensors* 2014, *14*, pp.7181–7208.

[35] Terroso M, Rosa N, Marques AT, Simons R. "Physical consequences of falls in the elderly: a literature review from 1995 to 2010", Eur Rev Aging Phys Act (2014)

[36] Available at http://www.healthinaging.org/aging-and-health-a-to-z/topic:falls/info :causes-and-symptoms/ [online] Accessed 15 November, 2017.

[37] Hormann W. Kommunikationstechnologie im Dienst amMenchen. In HAUSNOTRUFSYSTEME Kommunikationstechnologie imDienst am Menchen; Heilscher W, Ed, Wirtschaftsverlag NW: Bremerhaven, Germany, 1980. (In Geman)

[38] Hansen TR, Ekund JM, Sprinkle J, Bajcsy R, Sastry S. "Using smart sensors and a camera phone to detect and verify the fall of elderly persons", In Proceedings of the European Medicine, Biology and Engineering Conference, Prague, Czech Republic, 20-25 November 2005.

Appendix A

Source Code

```java
package com.example.n13631.final_project_try_006;

import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.support.annotation.IdRes;
import android.support.v7.app.AppCompatActivity;
import android.text.TextUtils;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.LinearLayout;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

    private static final String TAG = "Main_class";
    public static final String MyPREFERENCES = "f_p_prefs";
    SharedPreferences sharedprfs;
    public static final String phone1 = "phonekey1";

    String ph01,get_ph01;
    EditText num01;
    Button enter_button,btn_start_t_u,btn_start_m_u,btn_start_t;

    private RadioGroup first_group;
    private RadioGroup second_group;
    private RadioGroup third_group;
    private RadioButton minutes_r_btn, seconds_r_btn;
    private RadioButton shirt_r_btn,pant_r_btn,training_btn,non_training_btn;
    private TextView txt_ph_pos,txt_time_format;
    private Button btn_action_send;
    private EditText edt_txt_number;
    private LinearLayout ll;

    int ph_position_trn;

    boolean inshirt_pocket, inpant_pocket;
    boolean num_in_sec = false,num_in_min = false;
    boolean in_training = false;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        num01 = (EditText)findViewById(R.id.editText_num_01);
        sharedprfs = getSharedPreferences(MyPREFERENCES, Context.MODE_PRIVATE);
/* for storing the phone numbers */

        btn_start_t_u = (Button)findViewById(R.id.button_start_training_user);
        btn_start_m_u = (Button)findViewById(R.id.button_start_monitoring_user);
        ll = (LinearLayout)findViewById(R.id.layout_hidden);

        try{
            get_ph01 = sharedprfs.getString(phone1,null);
            if(!TextUtils.isEmpty(get_ph01)){
```

```
                num01.setText(get_ph01);
            }
        }
        catch (Exception e)
            e.printStackTrace();

        btn_start_m_u.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                ph01= num01.getText().toString();
                if(!ph01.equals("")&&ph01.length()==11) {
                    SharedPreferences.Editor editor = sharedprfs.edit();
                    editor.putString(phone1, ph01);
                    editor.commit();

                    Intent intent = new
Intent(getApplicationContext(),Comp_T_or_start_M.class);
                    startActivity(intent);
                    finish();
                }
            }
        });

        btn_start_t_u.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                setContentView(R.layout.user_choice_training);

                second_group = (RadioGroup) findViewById(R.id.radio_grp_phone_pos);
                third_group =
(RadioGroup)findViewById(R.id.radio_grp_time_selection);

                shirt_r_btn = (RadioButton)findViewById(R.id.r_btn_shirt_pocket);
                pant_r_btn = (RadioButton)findViewById(R.id.r_btn_pant_pocket);
                seconds_r_btn = (RadioButton)
findViewById(R.id.seconds_radio_button);
                minutes_r_btn = (RadioButton)
findViewById(R.id.minutes_radio_button);

                txt_ph_pos =
(TextView)findViewById(R.id.textView_trainig_phone_pos);
                txt_time_format =
(TextView)findViewById(R.id.textView_training_time);

                edt_txt_number = (EditText) findViewById(R.id.input_interval_time);
                btn_start_t = (Button)findViewById(R.id.button_start_training);

                second_group.setOnCheckedChangeListener(new
RadioGroup.OnCheckedChangeListener() {
                    @Override
                    public void onCheckedChanged(RadioGroup group, @IdRes int
checkedId) {
                        switch (checkedId){
                            case R.id.r_btn_shirt_pocket:
                                inshirt_pocket = true;
                                ph_position_trn = 1;
                                Log.e(TAG,"in shirt's pocket");
                                txt_time_format.setVisibility(View.VISIBLE);
                                third_group.setVisibility(View.VISIBLE);
                                break;
                            case R.id.r_btn_pant_pocket:
                                inpant_pocket = false;
                                ph_position_trn=2;
                                Log.e(TAG,"in pant's pocket");
                                txt_time_format.setVisibility(View.VISIBLE);
                                third_group.setVisibility(View.VISIBLE);
                                break;
                        }
```

```
                    }
                });

                third_group.setOnCheckedChangeListener(new
RadioGroup.OnCheckedChangeListener() {
                    @Override
                    public void onCheckedChanged(RadioGroup group, @IdRes int
checkedId) {
                        switch (checkedId){
                            case R.id.seconds_radio_button:
                                num_in_sec = true;
                                Log.e(TAG,"number format in sec");
                                edt_txt_number.setVisibility(View.VISIBLE);
                                break;
                            case R.id.minutes_radio_button:
                                num_in_min = true;
                                Log.e(TAG,"number format in sec");
                                edt_txt_number.setVisibility(View.VISIBLE);
                                break;
                        }
                    }
                });

                btn_start_t.setOnClickListener(new View.OnClickListener() {
                    @Override
                    public void onClick(View v) {
                        String getInterval =
edt_txt_number.getText().toString().trim();//get interval from edittext
                        int interval_inInt = 0;
                        try {
                            interval_inInt = getTimeInterval(getInterval);
                        } catch (Exception e) {
                            e.printStackTrace();
                        }
                        //check interval should not be empty and 0
                        if (!getInterval.equals("") && !getInterval.equals("0") &&
!(interval_inInt<60)){
                            Log.e(TAG,"in button clicked method- training");

                            Intent intent = new
Intent(getApplicationContext(),Acclerometer_v_for_training.class);
                            intent.putExtra("ph_position",ph_position_trn);
                            intent.putExtra("interval",interval_inInt);
                            startActivity(intent);
                            finish();
                        }
                    }
                });
            }
        });
    }

    //get time interval to trigger alarm manager
    private int getTimeInterval(String getInterval) {
        int interval = Integer.parseInt(getInterval);
        //Return interval on basis of radio button selection
        if (seconds_r_btn.isChecked())
            return interval;
        if (minutes_r_btn.isChecked())
            return interval * 60;//convert minute into seconds
        return 0;
    }

    @Override
    protected void onDestroy() {
        super.onDestroy();
    }
}
```