ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE

ENGINEERING AND TECHNOLOGY

STEP LENGTH ESTIMATION USING SENSOR FUSION FOR INDOOR POSITIONING

M.Sc. THESIS

Hasbi SEVİNÇ

Department of Computer Engineering

Computer Engineering Programme

JULY 2020

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE
ENGINEERING AND TECHNOLOGY

STEP LENGTH ESTIMATION USING SENSOR FUSION FOR INDOOR POSITIONING

M.Sc. THESIS

Hasbi SEVİNÇ
(504161518)

Department of Computer Engineering

Computer Engineering Programme

Thesis Advisor: Asst. Prof. Dr. Gökhan İNCE

JULY 2020

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

İÇ MEKAN KONUMLANDIRMASI İÇİN SENSOR FÜZYONU İLE ADIM UZUNLUĞU TAHMİNİ

YÜKSEK LİSANS TEZİ

Hasbi SEVİNÇ
(504161518)

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

Tez Danışmanı: Asst. Prof. Dr. Gökhan İNCE

TEMMUZ 2020

Hasbi SEVİNÇ, a M.Sc. student of ITU Graduate School of Science Engineering and Technology 504161518 successfully defended the thesis entitled "STEP LENGTH ESTIMATION USING SENSOR FUSION FOR INDOOR POSITIONING", which he/she prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor :	**Asst. Prof. Dr. Gökhan İNCE** Istanbul Technical University	....
Jury Members :	**Prof. Dr. Mustafa KAMAŞAK** Istanbul Technical University	....
	Asst. Prof. Dr. Mehmet Amaç GÜVENSAN Yıldız Technical University	....
		....

Date of Submission : **15 June 2020**
Date of Defense : **17 July 2020**

To my spouse and family

FOREWORD

First of all, I would like to thank my supervisor, Asst. Prof. Dr. Gökhan İnce, who provided all kinds of support for my thesis. I would also like to thank Uğur AYVAZ and Kadir ÖZLEM for their support in developing my thesis.

Finally, I would like to thank my spouse and my family who supported me at the points I had difficulty in my thesis and helped me complete my thesis.

July 2020

Hasbi SEVİNÇ
(Computer Engineer)

TABLE OF CONTENTS

Page

ABBREVIATIONS

CSV : Comma-separated Values
FFT : Fast Fourier Transform
GPS : Global Positioning System
GSM : Global System for Mobile Communications
IMU : Inertial Measurement Unit
MAC : Media Access Control
PCA : Principal Component Analysis
PDR : Pedestrian Dead Reckoning
QR : Quick Response
RFID : Radio Frequency Identification
RMS : Root Mean Square
SDK : Software Development Kit
UI : User Interface

SYMBOLS

$\mathbf{A_x}$, $\mathbf{A_y}$ $\mathbf{A_z}$	**:** Acceleration in *x*, *y*, *z* directions
$\mathbf{C_x}$, $\mathbf{C_y}$, $\mathbf{C_z}$	**:** Magnetic Field in *x*, *y*, *z* directions
$\mathbf{G_x}$, $\mathbf{G_y}$, $\mathbf{G_z}$	**:** Gyroscope in *x*, *y*, *z* directions
$\mathbf{R}^2$	**:** R-squared correlation

LIST OF TABLES

LIST OF FIGURES

Page

STEP LENGTH ESTIMATION USING SENSOR FUSION FOR INDOOR POSITIONING

SUMMARY

People use navigation applications to go to one place from another. Especially, if people traveling go to a place for the first time or if there is not enough information about the place, people traveling can get help from navigation applications. Navigation applications detect the person's location from Global Positioning System (GPS) or base station signals. However, the quality of these signals is not sufficient to use navigation applications in closed areas. In closed areas such as shopping centers, map information is usually provided through kiosk devices. The person looking for a store finds the store using the signs on the map at the mall. However, this is not possible for visually impaired people. Visually impaired individuals can reach their destination in open areas with navigation applications that use voice guidance. However, it is not possible to use these applications indoors, since they do not work properly in these locations. This study aims to provide navigation in indoor locations by using wearable and mobile devices.

Various studies have been carried out to provide navigation in indoor locations. These studies generally used wireless networks such as WiFi, Bluetooth, Radio Frequency Identification (RFID). The basic principle of these systems is the calculation of the distance to the network devices that emit signals. If the distance to three or more devices is known, the position of the person can be obtained. However, in these studies, some technical arrangements should be made in the building for the system to function properly. Therefore, indoor areas where navigation is provided with the help of signals are not widely available. In robotic studies, different methods have been developed for indoor location tracking. In these studies, robots generally find their positions according to predefined objects or Quick Response (QR) codes in the field. However, the robot needs to know the position of the objects in the environment to find its position. In the method presented in this thesis, it is aimed that there is no need for any installation inside the building to follow the position of the person.

For the study of the method presented in the thesis, the following elements were used: 1) textile-based capacitive sensors, 2) smart mobile phone, 3) WeWALK smart cane developed for the visually impaired. The data collection that is required for training of machine learning models were carried out with five different subjects walking on the established course. This track consists of walking paths with different stride lengths. Data taken from the sensors while the subjects walked in these areas were recorded by the system for further processing.

Textile-based capacitive sensors are placed in both knees of the subject. These sensors measure the angle changes in the subject's knee joint. Thus, the system obtains information about the steps that the subject takes while walking. Information about the characteristics of walking is obtained by using the accelerometer, gyroscope,

and the compass sensor in the smart cane and mobile phone. The direction of the walking subject was obtained from the compass sensor inside the mobile phone. As the first stage of the study, the data on the sensors were transmitted via Bluetooth connection. The system includes acceleration, gyroscope, and compass signals on x, y, and z axes for each of the smart canes and mobile phones. It also collects signals from the textile-based capacitive sensor. In total, the system collects nineteen signals. Firstly, the data collected from the sensors are cleaned from noise and outlier data using signal processing methods. An algorithm has been developed to detect the onset and offset points of the steps in the signal received from textile-based capacitive sensors. This algorithm calculates the local maximum and minimum points in the sensor signal and treats the interval between these points as steps. The signals in all sensors are segmented according to the determined start and finish points. In order to use sensor signals in regression models, feature extraction is performed on these signals. To improve the performance of the system, the extracted features are simplified by different feature selection methods. The performance of different methods were compared and regression models were trained with the best selection method. The system was trained with 1) Linear Regression, 2) Vector Support Regression, 3) Random Forest, 4) k-Nearest Neighbor models to find the best-fitted regression for the collected data set. The results obtained with these trained models were compared and the model with the best results was used for step detection. Sensor fusion was used to better determine stride lengths. In order to determine the contributions of the three different sensors in the system, different fusion alternatives have been tested separately and in pairs. As a result of these tests, it was observed that the fusion of three different sensors together provided the highest accuracy for step detection and lowest localization error.

An Android application using Google Maps has been developed to perform localization. First of all, the plan prepared with the actual dimensions of the test environment was loaded on the map of the application. The initial location of the person in the test environment is defined in the application and this location is shown on the map with a marker. The application uses the regression models developed to determine the step length. Sensor data collected via Bluetooth is subjected to the same signal processing methods as in the learning phase of the model. Thus, when the person starts walking, the application determines the step length of the person with the help of the model and it updates the marker on the map according to the person's direction information. In order to test the system performance, a track with the same origin and destination points was determined in the test environment. When the person completed the entire track starting from the initial point, the distance between their actual final position and the projected final position was calculated.

The method developed in this thesis aims to enable the visually impaired to reach their desired destinations in indoor locations by using the smart cane, textile-based capacitive sensor, and smartphones. With the data collected in walking tests, the development of regression models, and the Android application showing the position on the map, this study contributes to literature in the indoor localization and navigation.

İÇ MEKAN KONUMLANDIRMASI İÇİN SENSOR FÜZYONU İLE ADIM UZUNLUĞU TAHMİNİ

ÖZET

Günümüzde insanlar bir yerden bir yere gidebilmek için navigasyon uygulamalarını kullanmaktadır. Özellikle bir yere ilk defa gidiliyorsa veya yer hakkında yeterli bilgi yoksa navigasyon uygulamalarından yardım alınmaktadır. Navigasyon uygulamaları kişinin konumunu Global Positioning System (GPS) veya baz istasyonu sinyallerinden algılamaktadır. Fakat bu sinyallerin kalitesi kapalı alanlarda navigasyon uygulamalarını kullanmak için yeterli değildir. Alışveriş merkezleri gibi kapalı alanlarda harita bilgisi genellikle kiosk cihazları üzerinden sağlanır. Bir mağazayı arayan kişi, harita ve AVM içerisindeki tabelaları kullanarak mağazayı bulmaktadır. Fakat bu durum görme engelli kişiler için mümkün değildir. Görme engelli bireyler açık alanlarda, sesli yönlendirme özelliğine sahip navigasyon uygulamaları ile gitmek istedikleri yere ulaşabilirler. Fakat bu uygulamalar kapalı alanlarda yeterli doğrulukta çalışmadığı için kullanımı mümkün olmamaktadır. Bu çalışmada, giyilebilir ve mobil cihazlar kullanılarak, kapalı alanda navigasyonun sağlanması hedeflenmektedir.

Kapalı alanda navigasyonun sağlanması için çeşitli çalışmalar yapılmıştır. Bu çalışmalar genellikle WiFi, Bluetooth, Radio Frequency Identification (RFID) gibi kablosuz ağlar kurularak gerçekleştirilmiştir. Buradaki temel prensip sinyal yayan ağ cihazlarına olan uzaklığın hesaplanmasıdır. Eğer üç veya daha fazla cihaza olan uzaklık biliniyorsa, kişinin konumu hesaplanabilir. Fakat bu çalışmalarda sistemin doğru işlemesi için bina içerisinde bazı teknik düzenlemeler yapılmalıdır. Bu yüzden sinyal yardımıyla navigasyonun sağlandığı kapalı alanlar yaygın değildir. Robotik çalışmalarında da kapalı alanda konum takibi için farklı yöntemler geliştirilmiştir. Bu çalışmalarda genellikle robotlar, alan içerisindeki önceden tanımlanmış obje veya Quick Response (QR) kodlara göre kendi konumlarını bulmaktadır. Fakat robotun konumunu bulabilmesi için ortamdaki objelerin konumunu önceden bilmesi gerekir. Bu tez ile sunulan yöntemde, kişinin bina içerisindeki konumunu takip etmek için, bina içerisine herhangi bir kurulum ihtiyacının olmaması hedeflenmiştir.

Tezde sunulan kapalı alanda navigasyon sisteminin çalışması için kişinin ilk konumu bilinmelidir. Kişi bina içerisinde hareket ettiğinde kişinin adım uzunluğu makine öğrenmesi modelleri ile tahmin edilmektedir. Adım uzunluğunun tahmini için çeşitli çalışmalar yapılmıştır. Bu çalışmalarda ivme, manyetometre ve jiroskop sensörlerinden oluşan sistemler kullanılmıştır. Toplanan sensör verileri kişinin attığı adımların karakteristiği hakkında bilgi vermektedir. Kişinin doğrultusu da belirlendikten sonra kişi adım attığında adım uzunluğu tahmin edilerek bir önceki konumuna eklenmektedir. Böylece kişi yürürken eş zamanlı olarak kişinin güncel konumu tahmin edilebilmektedir.

Tezde sunulan yöntemin çalışması için 1) tekstil tabanlı kapasitif sensörler, 2) akıllı cep telefonu, 3) görme engelliler için geliştirilmiş WeWALK akıllı bastonu kullanılmıştır.

Makine öğrenmesi modellerinin eğitilmesi için gerekli olan veri toplama çalışmaları, kurulan parkur üzerinde yürüyen beş farklı denek ile yapılmıştır. Bu parkur farklı adım uzunluklarına sahip yürüme yollarından oluşmaktadır. Denekler bu alanlarda yürürken sensörlerden alınan veriler sistem tarafından daha sonra işlenmek üzere kaydedilmiştir.

Tekstil tabanlı kapasitif sensörler, bireyin her iki dizine yerleştirilmiştir. Bu sensörler bireyin diz eklemindeki açı değişimlerini ölçer. Böylece sistem, birey yürürken attığı adımlar hakkında bilgi sahibi olur. Akıllı baston ve cep telefonu içerisinde bulunan ivme, jiroskop ve pusula sensörünün kullanılmasıyla da yürümenin karakteristiğine dair bilgiler elde edilir. Kişinin yürüme doğrultusu, cep telefonu içerisinde bulunan pusula sensöründen elde edilir. Çalışmanın ilk aşaması olarak sensörlerdeki veriler Bluetooth bağlantısı üzerinden toplanmıştır. Sistem akıllı baston ve cep telefonundan her biri için, x, y ve z eksenlerinde ivme, jiroskop ve pusula sinyallerini; tekstil tabanlı kapasitif sensörden ise sağ veya sol diz için birer sinyal olmak üzere toplamda ondokuz sinyal toplamaktadır. Öncelikle sensörlerden toplanan data sinyal işleme metotlar ile birlikte gürültü ve hatalı datalardan temizlenmiştir. Tekstil tabanlı kapasitif sensörlerden alınan sinyalin içerisindeki adımların başlangıç ve bitiş noktalarının tespiti için bir algoritma geliştirilmiştir. Bu algoritma sensör sinyalindeki yerel maksimum ve minimum noktalarını hesaplayarak, bu noktalar arasında kalan kısımları adım olarak değerlendirir. Tüm sensörlerdeki sinyaller belirlenen adım başlangıç ve bitiş noktalarına göre kesilir. Sensör sinyallerinin regresyon modellerinde kullanılabilmesi için bu sinyallerden öznitelik çıkarımı yapılır. Sistemin performansını iyileştirmek için, çıkarılan öznitelikler farklı eleme yöntemleri ile sadeleştirilmiştir. Denenen farklı yöntemlerin sonuçları karşılaştırılmış ve en iyi eleme yöntemi ile regresyon modelleri eğitilmiştir. Toplanan veri kümesine en iyi uyan regresyon modelinin bulunması için 1) Lineer Regresyon, 2) Vektör Destek Regresyonu, 3) Rassal Orman, 4) k-En Yakın Komşu modelleri ile sistem eğitilmiştir. Eğitilen bu modeller ile alınan sonuçlar karşılaştırılmış ve en iyi sonuç veren model adım tespiti için kullanılmıştır. Çalışmada adımların daha iyi tespit edilmesi için sensör füzyonu kullanılmıştır. Sistemdeki üç farklı sensörün sisteme olan katkılarının belirlenmesi için sensör ayrı ayrı ve ikili olarak testlere tabii tutulmuştur. Bu testler sonucunda üç farklı sensörün birlikte değerlendirilmesi ile en doğru adım uzunluğu tahminin ve en az lokalizasyon hatasının yapıldığı gözlemlenmiştir.

Tezin hedefi olan kapalı alanda lokalizasyonun test edilebilmesi için Google Haritalarını kullanan bir Android uygulaması geliştirilmiştir. Öncelikle test ortamının gerçek ölçüleri ile hazırlanan plan, uygulamanın haritasına yüklenmiştir. Kişinin test ortamında bulunduğu ilk konum uygulamaya tanımlanmış ve bu konum harita üzerinde bir işaretçi ile gösterilmiştir. Uygulama adım uzunluğunu tespit edebilmek için geliştirilen regresyon modellerini kullanmaktadır. Bluetooth üzerinden toplanan sensör verileri, modelin öğrenme aşamasında olduğu gibi aynı sinyal işleme metodlarına tabii tutulmaktadır. Böylece kişi yürümeye başladığında, uygulama model yardımıyla kişinin adım uzunluklarını tespit etmekte ve kişinin yön bilgisine göre haritadaki işaretçiyi güncellemektedir.

Tez kapsamında geliştirilen sistemin performans ölçümlerinin yapılabilmesi için, başlangıç ve bitiş noktaları aynı olan bir test parkuru hazırlanmıştır. Bu test parkuru farklı adım uzunluklarından oluşmaktadır. Kişi başlangıç noktasından başlayarak tüm parkuru bitirdiğinde, olması gereken bitiş konumundan ne kadar uzaklıkta olduğu hesaplanmıştır. Sensör füzyonunun sisteme katkısını bulmak amacıyla testler farklı sensör kombinasyonları ile tekrarlanmıştır.

Bu tez ile geliştirilen yöntem, görme engellilerin kullandıkları akıllı baston, tekstil tabanlı kapasitif sensör ve akıllı cep telefonlarını kullanarak kapalı alanda gitmek istedikleri yere güvenle ulaşmalarını hedeflemektedir. Bu çalışma, yürüme testlerinde toplanan veri, regresyon modellerinin geliştirilmesi ve harita üzerindeki konumu gösteren Android uygulaması ile kapalı alan lokalizasyon ve navigasyon çalışmalarına katkı sağlamaktadır.

1. INTRODUCTION

In this section motivation of thesis, the aim of the study, and brief of the designed system are introduced.

1.1 Motivation

The daily environment humans live in is designed to make people comfortable. While organizing the environments, the needs of people are taken into consideration. However, these regulations are sometimes not sufficient for people with disabilities. People with disabilities face many problems in their daily lives such as climbing stairs, crossing the street, using public transport. With the development of technology, these deficiencies have started to be mitigated in many areas. For example, with the help of artificial intelligence, sign language can be converted into speech, and similarly, human speech can be converted into subtitles for video content [2, 3]. It is estimated that there are approximately 253 million intermediate or completely visually impaired people worldwide [4]. There are also many technological developments to make the life of the visually impaired easier. One of the most important developments in this area is the WeWalk[1] SmartCane. With this device, the visually impaired walking stick, which has remained to keep its shape and properties for centuries, has been adapted to the technology and made smart. Thanks to this smart cane, people can also be protected from obstacles in their body alignment. Also, thanks to the mobile application, the person can use the functions of the phone without taking the phone out of the pocket. One of the problems of the visually impaired is transportation. Today, transportation is carried out with navigation applications. When user tells the application where the user wants to go, the application can direct the user to the place with voice directives. Navigation systems often utilize Global Positioning System (GPS) satellites. However, these technologies cannot be used in most indoor areas. There are different localization methods developed for indoor areas. However,

[1] https://wewalk.io

to be able to use these methods, necessary installations must be made to the building. Therefore, these features cannot be used in all indoor areas. Regardless of the building, sensors and wearable technologies can be used to provide navigation indoors. Thus, the visually impaired person who uses these technologies can reach the place he wants to go safely in every indoor area as long as the map is provided in advance.

1.2 Purpose of the Thesis

Visually impaired individuals can go anywhere they want in outdoor areas by using navigation applications of smartphones. Even if these applications work with high accuracy, they do not work properly in closed areas such as shopping centers and faculties. In areas where GPS satellite connection cannot be established, the performance of these applications decreases. Using the proposed system in this study, it is aimed to help navigation systems to navigate visually impaired individuals safely. For this purpose, a new method has been proposed. In the developed method, the building does not need any additional technology such as WiFi or Bluetooth. The current position of the person was determined with the help of a wearable device, a smart cane, and phone sensors. The system calculates the current position by estimating the step lengths taken by the person. In addition to step lengths estimation, the direction of the person is also calculated with the help of sensors. Thus, when the first position is known, the user's position on the map can be updated when the person starts walking. With the integration of the developed system to existing navigation applications, people with visual impairments will be able to use their navigation applications in closed areas.

1.3 Overview of the Thesis

This thesis contains five chapters. In the Chapter 1, the motivation and goals of this study are explained. In the Chapter 2, current information about gait analysis, step length estimation methods, and different navigation systems are given. In the Chapter 3, the designed system for step length estimation is described. In the Chapter 4, the trained machine learning model is used in an Android application. This application has a map that shows the person's location. In the Chapter 5, performed experiments and their results are given. In the Chapter 6, the benefits of this study and possible future works to be done are discussed.

2. LITERATURE REVIEW

There are important studies about indoor navigation systems and gait analysis for detection and estimation. This chapter contains the previous studies about indoor navigation systems, step detection, and step length estimation systems. Methods, results, and reports of the studies will be discussed.

2.1 Gait Analysis and Step Detection

Gait is an important activity for human life. There are studies about gait analysis to help people who have walking disabilities. These studies analyze the movement dynamics of gait. Shahid et al. use a wearable suit named as Intelligent Gait Oscillation Detector [5]. The wearable suit measures joints of knees, hips, elbows, and shoulders of a walking person. This study shows changes of the joints in different subjects with different walking speeds. The main objective of this study is showing walking patterns to help studies that work on step detection. Chalmers et al. used a 3-axes accelerometer and 2-axes gyroscope sensors which are monitored by a microcontroller to analyze gait characteristics of children [6]. To detect gait characteristics, they have compared results of different algorithms on their sensor values. Their study proposes a wearable device that can be used for long-term foot-angle measurements.

Studies on gait analysis show that there are different phases of a gait. A walking step can be split into eight phases [7]. A full gait cycle contains initial contact, loading response, midstance, terminal stance, pre-swing, initial swing, mid-swing, and terminal swing phases. Pappas et al. use a gyroscope sensor to monitor the angular velocity of the foot [8]. Their system can detect gait step accurately by using a rule-based detection algorithm. The algorithm takes the velocity of the feet to decide which phase of the gait is being performed. Researches on gait analysis help studies on step detection systems. Step detection systems generally use a peak detection or zero crossing algorithms [9–12]. The accuracy of these methods depends on the positioning of the sensors. If sensors are attached to the required position of the body, they can count steps accurately. Otherwise, these methods suffer from the drift in the orientation. To overcome sensor positioning problem, Yang et al. presents a novel method by fusing accelerometer, magnetometer, and gyroscope signals [13]. In this study, the device reference frame is transformed into the earth reference frame. The newly calculated vertical acceleration is used with a peak detection algorithm.

2.2 Step Length Estimation Systems

Pedestrian Dead Reckoning(PDR) is an important research area for recently popular studies such as child tracking, tracking elderly people and recognizing their physical activities and indoor navigation of the visually impaired people [14–17]. For such application areas, Inertial Measurement Unit (IMU) sensor combinations such as an accelerometer, gyroscope, magnetometer, etc. are commonly used for PDR methods. A PDR algorithm that integrates WiFi fingerprint technology and smartphone sensors (accelerometer, gyroscope, barometer, magnetometer) is developed by Moder et al. [18]. They obtained the step, its length, and heading of the visually impaired pedestrians by using wifi signals and smartphone sensors. However, their system requires predefined knowledge about building and floor plan to navigate the pedestrian.

Renaudin et al. combined foot-mounted IMUs and handheld IMUs to better estimate step length. They introduced a novel approach to detect step length by using a handheld IMU sensor [19]. They also designed a validation mechanism using a wheel speed sensor handled by the pedestrian. Ho et al. proposed an adaptive step length estimation system using accelerometer data. They applied Fast Fourier Transform (FFT) smoothing algorithm to eliminate interference signals caused by the drift problem [20]. Alvarez et al. aimed to decrease the measurement errors caused by drift by combining multiple IMUs [21]. Their study has shown that the utilization of multiple sensors is a feasible approach that reduces drift errors compared to using a single sensor.

Shin et al. worked on an adaptive step length estimation algorithm [22]. The gait speed, and activity type of the person measured using Micro Electro Mechanical Systems. After the total number of steps is measured, the total distance traveled is calculated using the movement speed of the person. The purpose of their study was to calculate the calorie burn in gait. Therefore, they did not need to measure step length sensitively unlike we did in our research.

2.3 Navigation Systems

To position a person in the coordinate system, there are some services commonly used such as GPS and cellular network systems [23]. These systems obtain the current location by receiving signals from satellites or base stations of GSM operators.

The most common way of positioning is the GPS method. In this system, there are 24 satellites around the world. These satellites are transmitting signals which contain the time and location information of the satellite. A device that uses GPS to find its location receives these signals. The difference between the time information of the signal and receiving time of the signal makes the system to be able to calculate the distance to the satellite. When a device receives more than three satellite signals, it is possible to find latitude and longitude of the coordinate [24]. The accuracy of the GPS is in 5-30 meters for urban areas and 5 meters for rural areas [25].

Finding coordinates from a cellular network system uses the same method as GPS. It uses the time of base stations to find the position [26]. This localization system requires at least three base station signals for positioning. In urban areas, the accuracy of this system is 200 meters to 500 meters. In a rural area, accuracy can change 500 meters to 5 kilometers according to cell sizes of the cellular network [27].

Even if GPS used widely for outdoor applications which require high accuracy, it can not be used for indoor applications because of signal loss [28]. Therefore, there are studies about introducing new methods for indoor localization. These systems generally use signal generator hardware or some markers whose location is predefined.

The main problem of GPS is the signal loss in the interior of buildings. Therefore, Ozsoy et al. designed a GPS repeater hardware to overcome this problem [29]. They mounted this hardware on the roof of a building. Their system receives satellite signals from different parts of the sky. When a satellite signal is received, the signal is amplified by the hardware. The system retransmits the amplified signal into the building. Thus, a device that has the capability of receiving GPS signals can find its position in the building by using the amplified and retransmitted signals. But this method requires a hardware setup to make indoor positioning available in a building.

Instead of GPS, WiFi signals can be used for indoor navigation. Almost every building has a WiFi connection. A client such as a mobile phone can measure WiFi signal power to calculate the distance to the WiFi router by using the proximity algorithm [30]. If locations of WiFi routers are known by the client, it can obtain its position when more than one WiFi signal is received. To use this method, all locations in the building should be covered with WiFi signals. However, WiFi signals are easily affected by interference. This may reduce the accuracy of the localization inside the building.

Some stores and markets use beacons to advertise their products or analyze their customer behaviors [31]. Beacons generate Bluetooth signals and they communicate with clients' mobile phones. Shopping centers generally mount these gadgets near the entrances of stores and other important locations. It is also possible to use these beacon signals to find a client position in indoor areas [32]. The client measures Bluetooth signal strength to calculate its distance to beacon [33]. If beacon locations are known by the client, the client can navigate itself. To provide continuous navigation without any interruptions, the indoor area should have numerous of beacons.

Radio Frequency Identification(RFID) signals can be used for positioning. Chon et al. presented a system that uses RFID technology for positioning vehicles on roads [34]. They propose that RFID tags will be installed on roads. These tags will transmit signals which contain information about their location. When a vehicle passes over a tag, it will receive the RFID signal. In that way, the vehicle is able to obtain its location accurately. Xu at al. presented an algorithm which uses Bayesian probability and K-nearest neighbor methods to make RFID technology is able to use for indoor positioning [35]. To use this highly accurate indoor positioning technology in an area, lots of RFID tags need to be installed in the area. Therefore, it is a costly technology.

The indoor navigation topic is an important research area for robotics as well. In order to move robots from one place to another, some methods have been discovered. One of them attaches the encoder system to the robot's wheels [36]. When wheels turn, encoder detects changes and it sends pulses to the controller. The controller counts encoder pulses to calculate traveling distances. To apply this method, the client must have a wheel system. For small indoor areas, the client can detect some predefined landmarks to find its position [37–39]. When the client detects two or more landmarks with its camera, it calculates its location by measuring the distance to each landmark. This approach requires a lot of landmarks for large-sized indoor areas.

3. STEP LENGTH ESTIMATION SYSTEM

In this chapter, details of the step length estimation system are described. This system contains a machine learning model to estimate steps length while a person is walking. Firstly, the architecture of the system and used devices are introduced. Then signal processing steps and machine learning models are given.

3.1 System Architecture

GPS The step length estimation system was designed using sensor fusion. It fuses three different sensors. They are:

1. Textile-based capacitive sensors
2. IMU of a mobile phone
3. IMU sensor of WeWalk smartcane.

Textile-based capacitive sensors are attached to the knees of the person. They detect angle changes on knees. These changes give information about step phases. The onset and offset point of a step is detected by this sensor. IMU sensors of the smartcane and the mobile phone detect changes of accelerometer, gyroscope, and magnetometer data while walking. These changes give information about walking such as the direction of the person, walking acceleration in the x, y, and z axes.

Figure 3.1 shows the sensors used to collect data and the position of the sensors on the subject. Textile-based capacitive sensors are attached to knee pads that are worn by the person. It is important that the sensors must be on the knees. Otherwise, they can not give correct information about the changes in the angular position of the knees. The mobile phone is in the pocket of the person. The person holds smartcane and its stick which touches to ground.

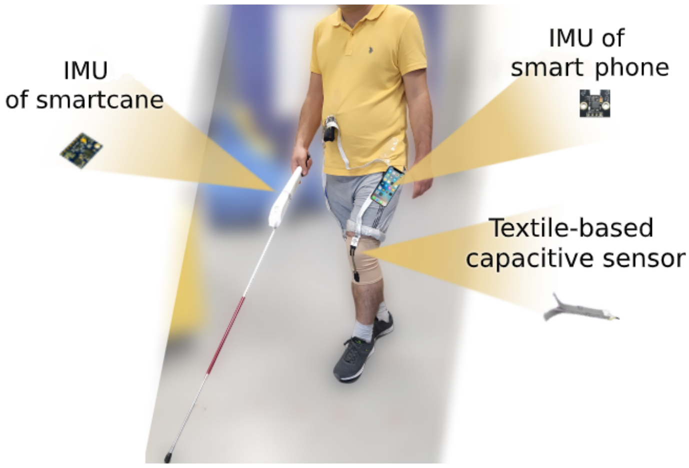

Figure 3.1 : Sensors used to collect data.

3.1.1 Textile-based capacitive sensors

Textile-based capacitive sensors are manufactured by adding an insulator silicone dielectric layer between two flexible highly conductive knitted fabrics [40]. When these sensors are attached to a brace in the knee joint, the capacitance value of the sensor changes with respect to the stretch changes on the joint during knee movements [41]. For example, when the bending angle of the knee increases during gait, the capacitance value also increases since the strain on the two ends of the sensor increases. Changes in the knee joint angle give information about the step length, i.e. if the step length is small the bending on the knee is also small and leads to low capacitance changes [1]. Figure 3.2 shows the textile-based capacitive sensor.

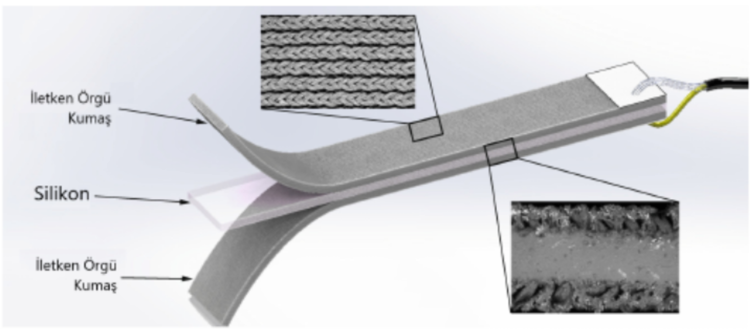

Figure 3.2 : Textile-based capacitive sensor [1].

Figure 3.3 shows the electrical circuit design of the sensor regarding the data collection diagram. In the circuit, all electronic components operate with 3.3 Volt. Therefore, a regulator is used to decrease battery voltage to 3.3 Volt. A microcontroller reads sensor values and it sends them to the Bluetooth module. When the circuit is powered on, the Bluetooth module starts with advertisement mode. In this mode, the Bluetooth module presents that it is ready for connection. The microcontroller should wait for "Bluetooth connected" signal before sending any data to the module. After the connection is established, the microcontroller periodically sends sensor data over the Bluetooth module.

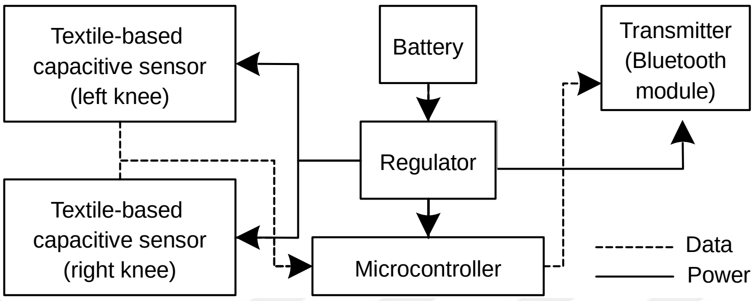

Figure 3.3 : Circuit block diagram used to collect textile-based capacitive sensor data.

3.1.2 Smartphone

A Smartphone is used to collect acceleration, gyroscope, and compass changes on the body during gait. Localization of smartphone on the person's body must be static while data collection for both training and testing. In the study, smartphone is put in the right pocket to measure the physical changes that occurred during gait. An Android application is developed to collect the sensor values with timestamps.

3.1.3 Smartcane

The smartcane "WeWalk" is developed for visually impaired. Figure 3.4 shows a picture of WeWalk. It has onboarded IMU sensors which can be used to measure physical changes during gait. Complementary to the smartphone, the smartcane allows us to measure the different dynamics of the walking movement as it contacts the ground.

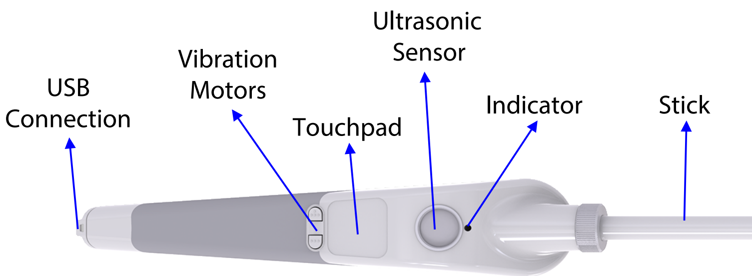

Figure 3.4 : WeWalk smartcane.

Besides, WeWalk performs important tasks for visually impaired people as a vibrating cane to warn the user when there is an obstacle, informing users when there is an incoming call, playing voice-activated directives, etc. Onboarded IMU sensor measures acceleration, gyroscope, and compass values in x, y, and z axes. When measurements are ready, the microcontroller of the smartcane sends these measurements to the mobile phone. It uses a Bluetooth connection to communicate with mobile phones. It has its own Software Development Kit(SDK) to manage connection and communication progress. Developed iOS or Android applications can get sensor values periodically by using this SDK.

3.2 Data Acquisition

There are three different sensor components that need to be collected for step length estimation. To detect step length correctly, all sensor data must be collected at the same time. If the synchronization of data is corrupted, sensor data represents an incorrect phase of a step. As a consequence, the accuracy of machine learning models will be reduced. There are two ways of data collection. One of them is sending data to a central device such as a computer. In this method, a device is selected as central. This central device has Bluetooth connectivity and all sensor components connect to the central device to send their data. Central device stores sensor data with its local clock time to prevent data synchronization. The second method of data collection is storing sensor data in the internal memory of the sensor component. In this method, to prevent data synchronization, all data should be stored with a timestamp which is generated by a real-time clock component. Before collection, real-time clocks of all sensors must be set at the same time.

In the proposed system, all three sensor components can send their data via Bluetooth connection; but textile-based capacitive sensors do not have any internal memory to store their data. Therefore this sensor should send its data to the central device. While collecting data, monitoring can be performed simultaneously. To monitor sensor data, a computer that has Bluetooth connectivity is selected as the central device. On the other hand, smartcane connects to mobile phones instead of computers. Moreover, phone IMU sensor data can be stored in the internal memory of the phone to reduce the Bluetooth communication load. Therefore, a hybrid method is selected as:

- The textile-based capacitive sensors will send its data to the computer and the computer will store this data with computer clock time.
- Smartcane connects to the mobile phone and the mobile phone will store sensor data with its clock time.
- IMU sensor data of the mobile phone is stored in the internal memory with mobile phone clock time.

To enable data time synchronization, clock times of the computer and the mobile phone are set to the same time in millisecond sensitivity. This technique is only used for data collection which is for the machine learning model. While performing step length in the processing pipeline, all sensor data will be collected by the mobile phone. The data transfer rate of each device is determined as 50 Hz.

Sensors and the central device communicate with each other in string format via Bluetooth. A communication packet starts with an exclamation mark and it continues with sensor data. Each sensor data is separated from the next with a comma. Every packet ends with a number sign and end-line character. "!720;751#" is a sample data packet where 720 shows left textile-based capacitive sensor value and 751 shows right textile-based capacitive sensor.

A NodeJS script is written to store textile-based capacitive sensor data. This script creates a Bluetooth server to establish a connection between the sensor and the computer. When data is received, current clock time and the data are stored in a CSV file. At the same time, data is shown in real-time graphs. The operator who manages data collection can check the status of the data collection by monitoring these graphs. Figure 3.5 shows the developed nodeJS script.

An Android application is developed to get IMU data from the mobile phone. This application also connects to the smartcane to get its IMU data. Both IMU data are stored in a CSV file with a timestamp.

3.3 Signal Processing

Collected data is not suitable to train machine learning models directly. Therefore there are some signal processing methods which must be applied such as smoothing

Figure 3.5 : A screenshot from NodeJS script during data acquisition.

the signals, removing the outliers, etc. Electronic circuits may misread the sensor values. Therefore, outlier data is removed from signals on the fly. Then, sensor signals are smoothed by a low pass filter. Smoothed data is ready for step detection. When a step is detected, feature extraction is applied to the step signals. These extracted features contain information about signal characteristics. To increase the performance of the system, dimensional reduction methods are also applied. These methods reduce the dimension of the feature set by removing or combining features. By using these concentrated features, machine learning models can be trained or used to predict step lengths. Figure 3.6 shows an overview of signal processing steps.

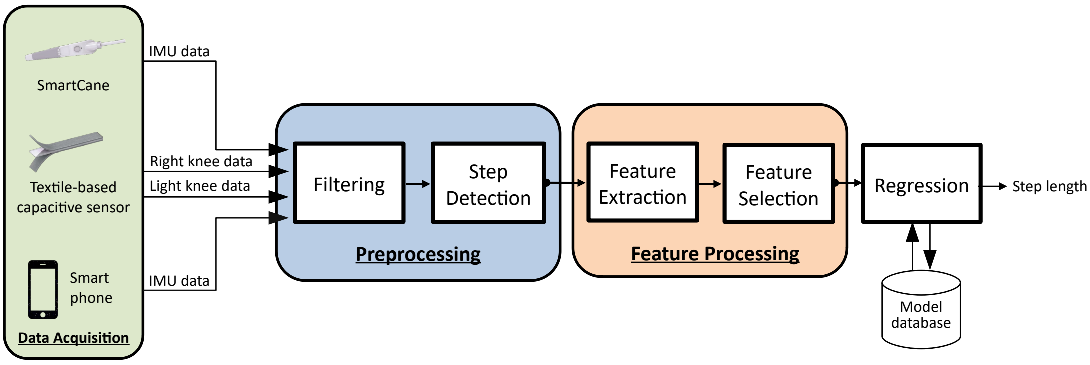

Figure 3.6 : Overview of the signal processing for step length estimation.

3.3.1 Preprocessing

Before using a signal, there are some methods to make the signal quality better. Applied preprocessing methods are listed below.

3.3.1.1 Outlier elimination

Electrical systems mostly generate noisy data. A temporary loss contact in the cables between sensors and microcontrollers may add noise into the signal, or sensor itself may misread the value of the physical world. Therefore, a maximum and a minimum value are set for all sensor signals. If a signal value is greater or lower than the threshold, the system decides it as an outlier and this value is removed from the signal. If this outlier signal is not removed from the signal, it may affect all systems to work incorrectly. Deciding the minimum and maximum threshold values is important. If points are set incorrectly, useful values of the signal may be removed.

3.3.1.2 Smoothing

A pure signal may contain short-term fluctuations. These fluctuations cause false-positive decisions in step detection methods because the methods suppose these fluctuations as local minima or maxima points. To remove short-term fluctuations and to get only long-term trends, a Butterworth low pass filter is applied to all sensor signals. The order of the filter is selected 5, the sample rate is selected as 50 Hz which is the frequency of collecting data acquisition, and the cutoff frequency is selected as 25 Hz.

3.3.1.3 Normalization

The value of the different sensors can be in different ranges. For example, in the system, textile-based capacitive sensors generate signals in 600-800 range, on the other hand, IMU values are generally between -3 and 3. To prevent signals from contributing to the machine learning models in an unbalanced manner to the system, high-value dominant data that may occur instantly must be normalized. Therefore, all sensor signals were normalized between 0 and 1.

3.3.1.4 Step detection

Step detection is an important phase of the signal processing pipeline, because, after obtaining the onset and offset point of the step, feature extraction and step length estimation will be performed. Depending on the position of feet and heels, there are seven phases of a full gait.

Phases of a gait are the following:

- Initial contact: Starting phase of a gait
- Opposite toe-off: The first-time left foot does not touch to ground
- Heel rise: The first-time left heel does not touch to ground
- Opposite initial contact: Middle phase of gait that both feet touch to the ground
- Toe off: The first-time right foot does not touch to ground
- Feet adjacent: Both feet at the same position
- Tibia vertical: Right foot is vertical

Textile-based capacitive sensors that are attached to knees give information about gait phases. When a foot started from the initial position to the middle phase, the knee angle is increasing. This angle change also increases the values of the attached textile-based capacitive sensors. After the middle phase of the gait, the knee angle and sensor values both decrease back to the initial value. The system detects a step according to these periodic changes on the sensor values.

To detect a step, the system marks all local minima and maxima points. When a local maxima point is detected, the system searches the local minima just before the local maxima. If a local minimum is not found, the system waits for the next local maxima. If there is a local minima point before a local maxima point, the system waits for the next local minima point. When it is detected, it means a gait is completed. These phases are repeated to detect all steps while walking. Pseudocode in Algorithm 1 shows the step detection algorithm of the system.

Algorithm 1: Step detection algorithm

```
Result: detected step signal
while True do
    m <- detect local maxima point;
    if m is not null then
        a1 <- get previous local minima;
        go back to the onset of the loop;
    else
        wait until next local minima;
        a2 <- get local minima;
        the onset time of gait <- the time of a1;
        the offset time of gait <- the time of a2;
        trigger step detected function with times;
    end
end
```

In Figure 3.7, a diagram of gait data regarding 6 right and 6 left steps.

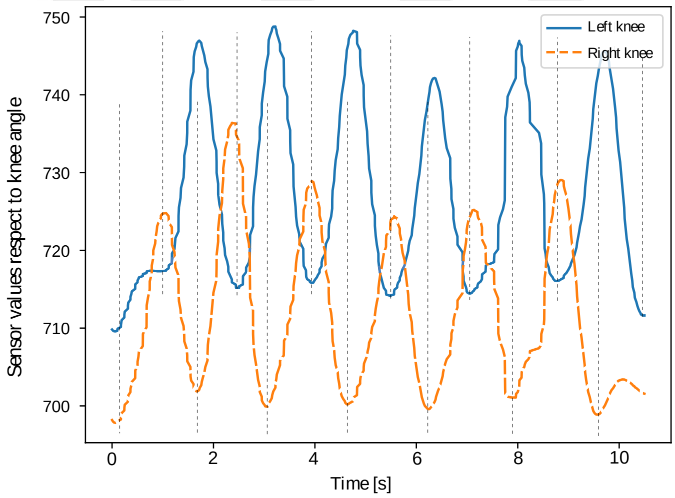

Figure 3.7 : Gait data and detected step segments.

3.3.2 Feature extraction

Before the machine learning model to work with high accuracy, the features to be extracted from the signal are of great importance. These extracted features should present the characteristic of the signal. Various feature extraction studies have been conducted on this subject [42, 43]. A weak presentation of the signal characteristics may cause false decision by machine learning models. A new function that takes a signal as an array is described. This function extracts statistical variables from the array. From each detected step signal, 13 features were extracted for both right and left knee from the capacitive sensor signals. Table 3.1 shows the extracted features and their descriptions.

Table 3.1 : Extracted features and their descriptions.

Feature	Definition
Min	the minimum value in a step sequence
Max	the maximum value in a step sequence
Mean	average of values in a sequence
Variance	how far values of a sequence are spread out from their average
Standard deviation	amount of variation of a sequence
Median	the middle value in a sequence
Entropy	uncertainty in a sequence
Interquartile range	variability value on each quartile of a sequence
Kurtosis	difference between tails of the distribution and a normal distribution
Skewness	measure asymmetry degree of a distribution
Mean absolute deviation	the average distance between the mean and each value
Range	difference between the smallest and largest values in a sequence
Mode	a value that repeated more often than any other

These variables are stored in a CSV file as a feature set. In addition to these features, the same features were also extracted from the smartphone and smartcane IMUs having three sensors (accelerometer, gyroscope, and magnetometer) with *x*, *y*, and *z* signal sequences. The total features count is calculated as follows; 2x13 from capacitive sensors for right knee and left knee, 3x3x13 from smartphone IMU, and 3x3x13 from smartcane IMU. In total, the dataset contains 260 features.

3.3.3 Feature selection

The quality of the extracted feature set is important for the performance of machine learning models. Irrelevant features increase the complexity of models and this may cause a reduction in accuracy score. There are some feature selection methods to reduce complexity. Commonly used backward feature elimination and principal component analysis (PCA) techniques were selected as feature selection methods [44, 45]. The backward feature elimination technique starts with all features in the feature set. It removes a feature from the feature set and runs the machine learning model to get the performance of the system. If the performance of the system was not increased, the feature is restored to the feature set. This step is repeated to find the best performance on the model. On the other hand, PCA does not remove a feature from feature set directly. It combines features and dimensionally reduced feature set. In this technique benefits of all features is prevented. Performance changes of the dimension reduction techniques are investigated in Chapter 5.

3.4 Regression

The machine learning model which takes features of step signals will predict step lengths. A machine learning model examines data of previous evidence to predict the result of new evidence. Therefore, the model should be trained with a feature dataset.

In preliminary phases of the system proposal, classification methods are used for step length estimation. To do this, all feature sets are labeled with its actual step lengths such as 10, 20, 30 cm, etc. Support vector machine, k-nearest neighborhood, decision tree, and stochastic gradient descent are selected as classifier methods. However, during natural walking, step lengths are not always equal to a fixed step size (e.g. 10, 20, 30 cm, etc). Since they will not be equal to the labeled lengths, models should be trained with a regression approach instead of category-based methods in step length calculation studies. Thus, instead of classifier methods, regression methods such as 1) Linear Regression, 2) Support Vector Regression, 3) k-Nearest Neighbor, and 4) Random Forest are used. These models were trained with training dataset and their performance on unseen test data set were compared. The hyperparameters of these regression models are shown in Table 3.2. As a result, the performance of each machine learning model has been tested and the most successful model has been selected for step estimation.

Table 3.2 : Hyperparameters of regression models.

Model	Hyperparameter	Value
	Kernel	Lasso regression (L1)
Linear Regression	Alpha	0.0001
	Elastic net mixing	0.5 L1 and 0.5 L2
	Kernel	Sigmoid
Support Vector	Cost (C)	3.00
Regression	Complexity bound (v)	0.5
	Iteration limit	1000
	Number of threes	10
Random Forest	#Attributes at each split	20
	Subsets lower limit	5
	Number of neighbors	5
k-Nearest Neighbor	Weight	Distance
	Distance metric	Manhattan

4. REAL-TIME INDOOR NAVIGATION SYSTEM

In this section, an android application was designed to establish a navigation system in real-time with indoor localization.

4.1 System Overview

An Android application is developed to track the person on a map by using the developed regression model. This application connects sensors via Bluetooth to get their measurements. It also reads accelerometer, gyroscope, and compass sensor values of the smartphone. First of all, the application stores the sensor data with timestamps. The step detection algorithm is called after preprocessing methods are applied. When a step is detected, features are extracted from all sensor signals of that step. According to the training model, the simplified features are used for the regression. Then the length of the step is obtained with the help of the regression model. It is assumed that the application is provided with the initial position of the person. After the direction is obtained with the compass sensor, the movement of the person is calculated and the marker that shows the location of the person is updated. Thus, the location on the map is updated in real-time while the person is walking. The architecture of the system is shown in Figure 4.1.

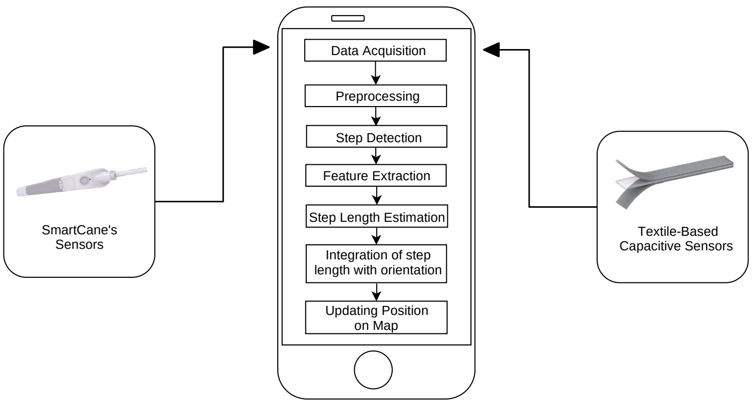

Figure 4.1 : Architecture of developed mobile application.

4.2 Mobile Application for Indoor Navigation

4.2.1 Application requirements

- The Android operating system requires location and Bluetooth permission to search and connect a Bluetooth device. Therefore, application should request this permission when it is launched for the first time.
- Application must check Bluetooth status. If it is switched off, the application must turn it on.
- The application must store sensor data with timestamps to provide data synchronization.
- Signal processing methods used in training regression models must be applied with the same parameters and in the same order.
- The application must have a map with a marker that shows the location of the person.
- The marker must be manipulated with only the developed system. Therefore, all other location services such as GPS must be turned off.
- The application should store test data to be able to compare results later.

4.2.2 Bluetooth connection between sensors

Bluetooth connections are established on the first two screens of the application. These screens share the same User Interface(UI) layout. The aims of the first and the second pages are connecting the smartcane and textile-based capacitive sensors, respectively. Connection pages are shown in Figure 4.2 where the screen on the left shows connection page of smartcane and the screen on the right shows connection page of textile-based capacitive sensor. When these pages are launched, the Bluetooth discovery service of the Android operating system is called. Before using any Bluetooth system functions, the user must give Bluetooth and location permission to the application. Otherwise, the application can not find any Bluetooth device. When the search is completed, the operating system triggers a callback. The application

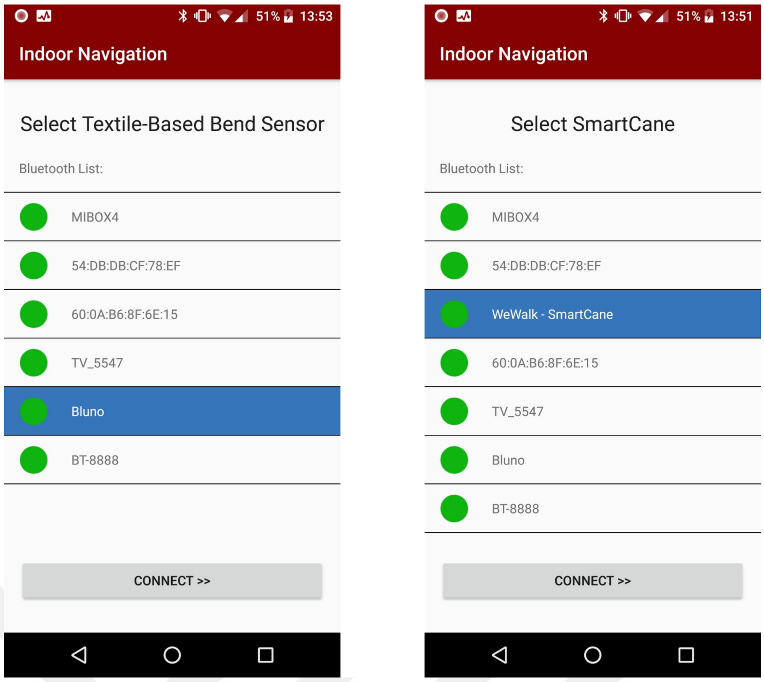

Figure 4.2 : Connection pages of application.

listens to the callback and it stores these discovered devices as a list on the screen. The user selects the smartcane or textile-based capacitive sensors from the list to establish a new connection. The application sends a connection request to the operating system with the selected device's Media Access Control(MAC) address. When the connection is established successfully, the operating system returns Bluetooth handler objects to the application. The application should store these connection objects to communicate with connected Bluetooth devices later.

4.2.3 Mapping

When all required Bluetooth connections are established successfully, the application launches the third screen. This screen contains a map to indicate user location. The map was created by using Google Maps library because it is free and easy to use with Android applications. Figure 4.3 shows integrated Google Map into the application. The walls on the map were drawn with the help of Google Custom Maps Service. In the figure, red marker shows position and direction of the person. When Google Maps is initialized for the application, the predefined initial location is marked and the map zooms towards the mark. When a new step is detected, the application calculates a new coordinate to update marker on the map. To obtain the new coordinates, step length, and direction data gets added to the person's previous coordinates.

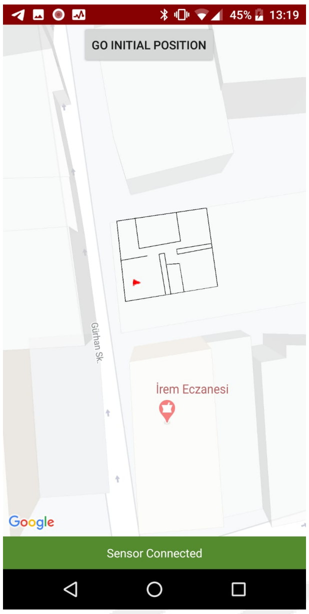

Figure 4.3 : Integrated Google Map into the application.

4.2.4 Update of the map

Google Maps works with latitude and longitude coordinate system. Latitude shows the x-axis and longitude shows the y-axis of a coordinate. Creating a new coordinate is done by adding latitude and longitude differences to the current location. Converting step length into coordinate difference is calculated[1] by Equation (4.1) and (4.2). This formula requires current latitude and longitude as la and lo, respectively, movement length as L and direction of movement as D. ER shows constant value of earth radius that is equal to 6371009.

$$Latitude = \arcsin(\cos(\frac{L}{ER}) \times \sin(la) + \sin(\frac{L}{ER}) \times \cos(la) \times \cos(D)) \tag{4.1}$$

$$Longitude = \arctan(\frac{\sin(\frac{L}{ER}) \times \cos(la) \times \sin(D)}{\cos(\frac{L}{ER}) - \sin(la) \times \cos(\frac{L}{ER})} \times \frac{1}{\sin(la) + \sin(\frac{L}{ER}) \times \cos(la) \times \cos(D)}) + lo \tag{4.2}$$

The direction of the person should be obtained in the middle phase of the walking step. Otherwise, when a person is turning, the direction may be reflected incorrectly by the

[1]https://www.edwilliams.org/avform.htm

map. The middle point of the step can be obtained by getting the peak point of the textile-based capacitive sensor signal. When the sensor signal reaches the peak point, it is stored with a timestamp. Then, this timestamp is used to get the exact direction value from the direction array which also stores timestamps.

4.3 Application Workflow

4.3.1 Preprogressing data

Data is received from the Bluetooth module and it stores an array with timestamps. Timestamp provides data synchronization between different devices' data. In the first step, the application applies a low pass filter to get smoother data and it eliminates outlier data which is not in the sensor value interval. Pre-progressed data is stored in different data arrays.

4.3.2 Step detection

Steps are detected by using the algorithm defined in Section 3.3.1.4. Sensor arrays are split into subarrays by the onset and offset time of the step. Thus, the application has sensor value arrays for the step.

4.3.3 Feature extraction

A feature extraction function is programmed in Java with the same parameters as Python script, which was used to train machine learning models. This function gets the data array of a sensor and it returns a feature object. This progress is repeated for all sensors. The dimension of extracted features must be equal to the dimension of the extracted feature list while model training. Otherwise, the application will fail to run the machine learning model.

4.3.4 Applying machine learning model

Hyperparameters of the machine learning model are obtained in Python in Section 3.4. This Python script reads the dataset and trains machine learning and gets model results, but this output is not suitable for the Android application. To use the model in the application, Weka software is used. The same dataset is applied to Weka software with the same machine learning model and its hyperparameters. When training is completed, the machine learning model is exported as a Weka model. The Weka software is written in Java, thanks to Weka Android library, it is easy to use Weka models in Android applications. The output of feature extraction is applied to the machine learning model by using the Weka model functions. This function returns the estimated step length.

5. Experiments and Results

In this section, information about experiment setup and evaluation metrics were given. First, step length estimation experiments, then navigation experiments were carried out. The results of these experiments were shared and compared in Sections 5.1.3 and 5.2.

5.1 Step Length Estimation Experiments and Results

5.1.1 Experimental setup

A track is set up to collect sensor data to create a dataset. This track consists of 20, 30, 40, 50, 60 cm step lengths. A drawing of the track is shown in Figure 5.1. Data is collected from 5 different volunteers walking at a steady speed on the track. Each person wore sensors and they walked the whole track 20 times. Sensor measurements were collected via Bluetooth and labeled with corresponding step length and timestamps. The created dataset contains equal test counts for each label. It was divided into 72%, 18%, and 10%, as training, test, and validation data sets, respectively. There were three sensors acquiring data during natural walking. To investigate the effect of sensor fusion, firstly, sensor processing will be carried out individually. Then, the performance of the proposed sensor fusion system will be inspected.

Orange toolkit[1] is used to train regression models and evaluate the results. This software takes a dataset as input and it trains machine learning models with selected hyperparameters. Weka[2] framework is used to provide an interface to use a trained regression model in Android application. Bluetooth connection of textile-based

[1] https://orange.biolab.si
[2] https://www.cs.waikato.ac.nz/ml/weka

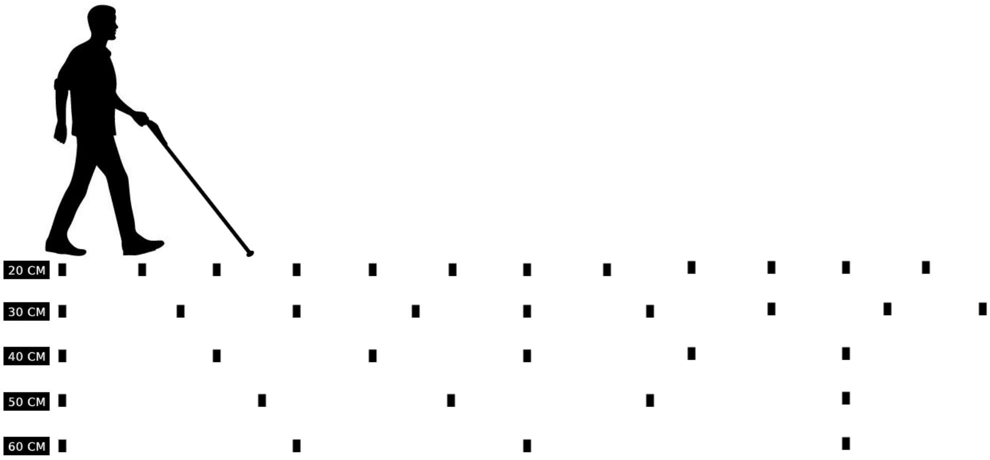

Figure 5.1 : Illustration of the track the test subjects walked on.

capacitive sensors is established by Bluno Beetle[3], which creates a Bluetooth server to send sensory data to the client. Bluno is programmed by using Arduino[4] functions.

5.1.2 Evaluation metrics

The system should work with a machine learning model that shows highest performance on the dataset. Used evaluation metrics and their formulas are given below.

$\boldsymbol{R^2}$ calculates the difference between the predicted and actual value. The difference is called residual. Squares of residuals are summed. Equation (5.1) shows the formula of R2. In the formula x shows predicted values, y shows the actual value, i shows index of summation and N shows the total number of the data. When R2 value is close to 1 that means the trained model is compatible with the dataset. When the dataset is not compatible with the dataset, R2 values get close to 0.

$$R^2 = \left(\frac{N(\sum_{i=1}^{N} x_i y_i) - (\sum_{i=1}^{N} x_i)(\sum_{i=1}^{N} y_i)}{\sqrt{[N\sum_{i=1}^{N} x_i^2 - (\sum_{i=1}^{N} x_i)^2][N\sum_{i=1}^{N} y_i^2 - (\sum_{i=1}^{N} y_i)^2]}}\right)^2 \tag{5.1}$$

Root Mean Square (RMS) is described as the square root of the mean squared error. Same as MSE, x shows predicted values, y shows the actual value, i shows index of summation, and N shows the total number of the data. Equation (5.2) shows the

[3]https://www.dfrobot.com/product-1259.html
[4]https://www.arduino.cc

formula of RMS.

$$RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - y_i)^2} \tag{5.2}$$

5.1.3 Results

In this section, the performance of the models in terms of error are presented. PCA method has been applied to reduce the size of the extracted features to 50 features. This way, the performance and operating speed of the system are increased by reducing the complexity. Models were trained using the features obtained with PCA and the results of these trained models are compared in Table 5.1.

Table 5.1 : Error performance of different regression models of the proposed system.

Method		RMS	R^2
Linear	*wo. PCA*	3.433	0.941
Regression	*w. PCA*	3.371	0.943
Support Vector	*wo. PCA*	8.531	0.636
Regression	*w. PCA*	4.119	0.915
k-Nearest	*wo. PCA*	9.291	0.568
Neighbor	*w. PCA*	2.371	0.972
Random	*wo. PCA*	2.790	0.961
Forest	*w. PCA*	3.731	0.93

The best score was obtained with PCA on KNN model. It increased accuracy on regression models except for Random Forest. PCA technique will continue to be applied in the rest of the regression models. Figure 5.2 and 5.3 show scatter graphs of models. The X-axis shows estimated step lengths and the y-axis shows ground truth values.

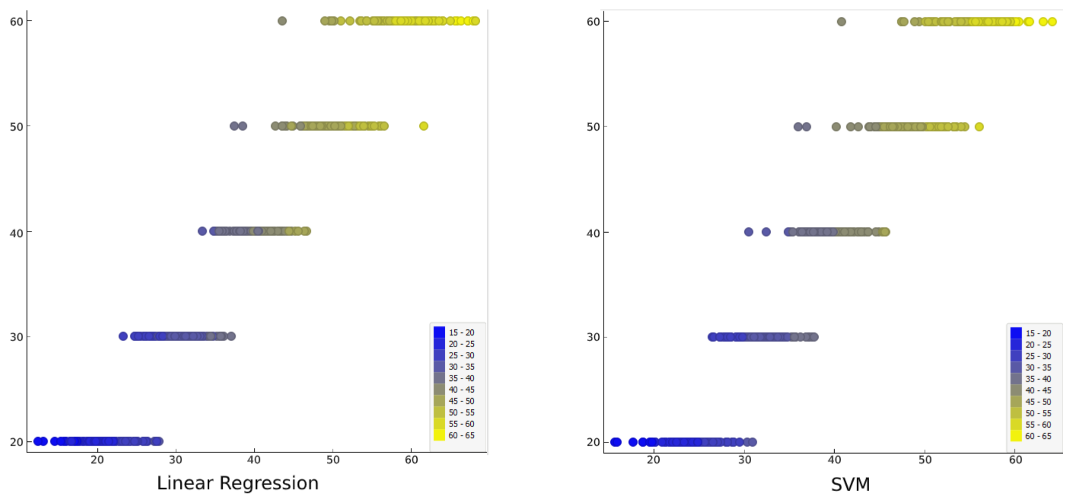

Figure 5.2 : Scatter graphs of Linear Regression and Support Vector Regression Models.

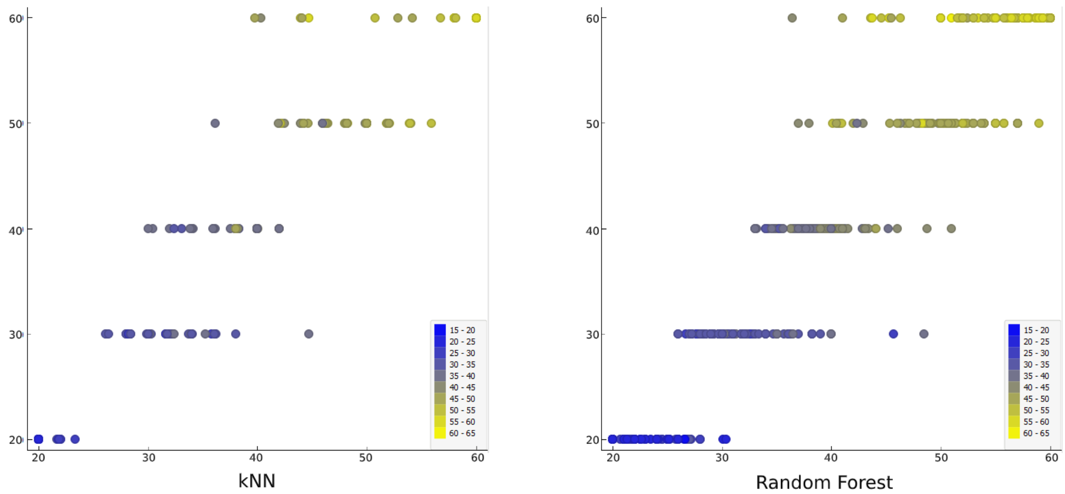

Figure 5.3 : Scatter graphs of k-Nearest Neighbor and Random Forest Models.

Table 5.2 shows R^2 results and Table 5.3 shows *Root Mean Square* scores of sensor fusion. Sensors were used with different combinations to find best combination for regression models. Firstly, all sensors were used for step length estimations separately. Mobile phone demonstrated highest performance among individual sensors with 0.961 R^2 score. Secondly, two sensors were combined together. Comparing with using only one sensor, two different sensor fusion improved models' scores. Fusion of mobile phone's sensor and textile-based capacitive sensors achieved the best score with 0.966 R^2 in the two sensor fusion variations. Lastly, we combined all three sensors to build the most comprehensive feature set. According to these results, the most promising method applied on test dataset was k-nearest neighbor regression method with 0.972 R^2 score.

Table 5.2 : R^2 scores of different combinations used in sensor fusion.

Sensor	LR	SVR	kNN	RF
Smart Cane	0.889	0.846	0.909	0.848
Smart Phone	0.899	0.889	0.961	0.934
Textile	0.729	0.510	0.845	0.807
Smart Cane + Textile	0.897	0.869	0.934	0.895
Smart Phone + Textile	0.917	0.874	0.966	0.934
Proposed System	0.943	0.915	0.972	0.930

Table 5.3 : Root Mean Square scores of different combinations used in sensor fusion.

Sensor	LR	SVR	kNN	RF
Smart Cane	4.719	5.544	4.258	5.511
Smart Phone	4.505	4.718	2.504	3.633
Textile	7.366	9.895	5.566	6.212
Smart Cane + Textile	4.547	5.112	3.626	4.579
Mobile Phone + Textile	4.075	5.014	2.604	3.62
Proposed System	3.371	4.119	2.371	3.731

5.2 Navigation Experiments and Results

5.2.1 Experimental framework

A track was prepared in a realistic home environment to carry out the tests with the Android application. This track consists of 68 arbitrary and consequent steps with random lengths. The subject starts from the beginning of the track, follows the course path and returns to the starting point. In Figure 5.4, red line shows track path and blue arrow shows starting point of the track.

Figure 5.4 : Sketch map of the track.

Figure 5.5 shows a photo from the navigation experiment where the application screen shows position of the person in real-time and black points in the photo shows ground-truth step points. In the experiment an Android phone with 4 GB RAM, Qualcomm Snapdragon 630 CPU, Android 8.0 OS was used.

5.2.2 Results

Figure 5.6 shows trajectories of different sensor fusion approaches. In the trajectory, dashed lines show ground truth and each trajectory starts from 0 points in x and y axes

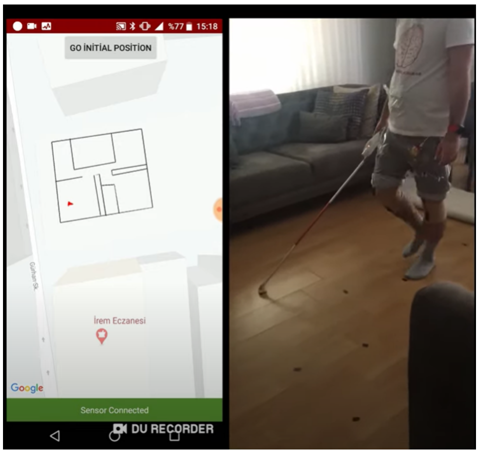

Figure 5.5 : A photo from the navigation experiment.

that show coordinates in cm. The orientation of the person is obtained by sensors of the mobile phone. Therefore, cumulative errors on the orientation data cause drifting on ground-truth and other trajectories.

The estimated step length by each sensor fusion approach is subtracted from corresponding ground truth value. Figure 5.7 shows the cumulative error (y-axis) after corresponding number of steps (x-axis). The proposed system made 1.45-meter cumulative error on the 20-meter long experiment track.

These results show that the closest path to ground truth is obtained by our proposed system relying on textile-based capacitive sensor, smartphone and mobile phone fusion. The second-best result was obtained with the mobile phone.

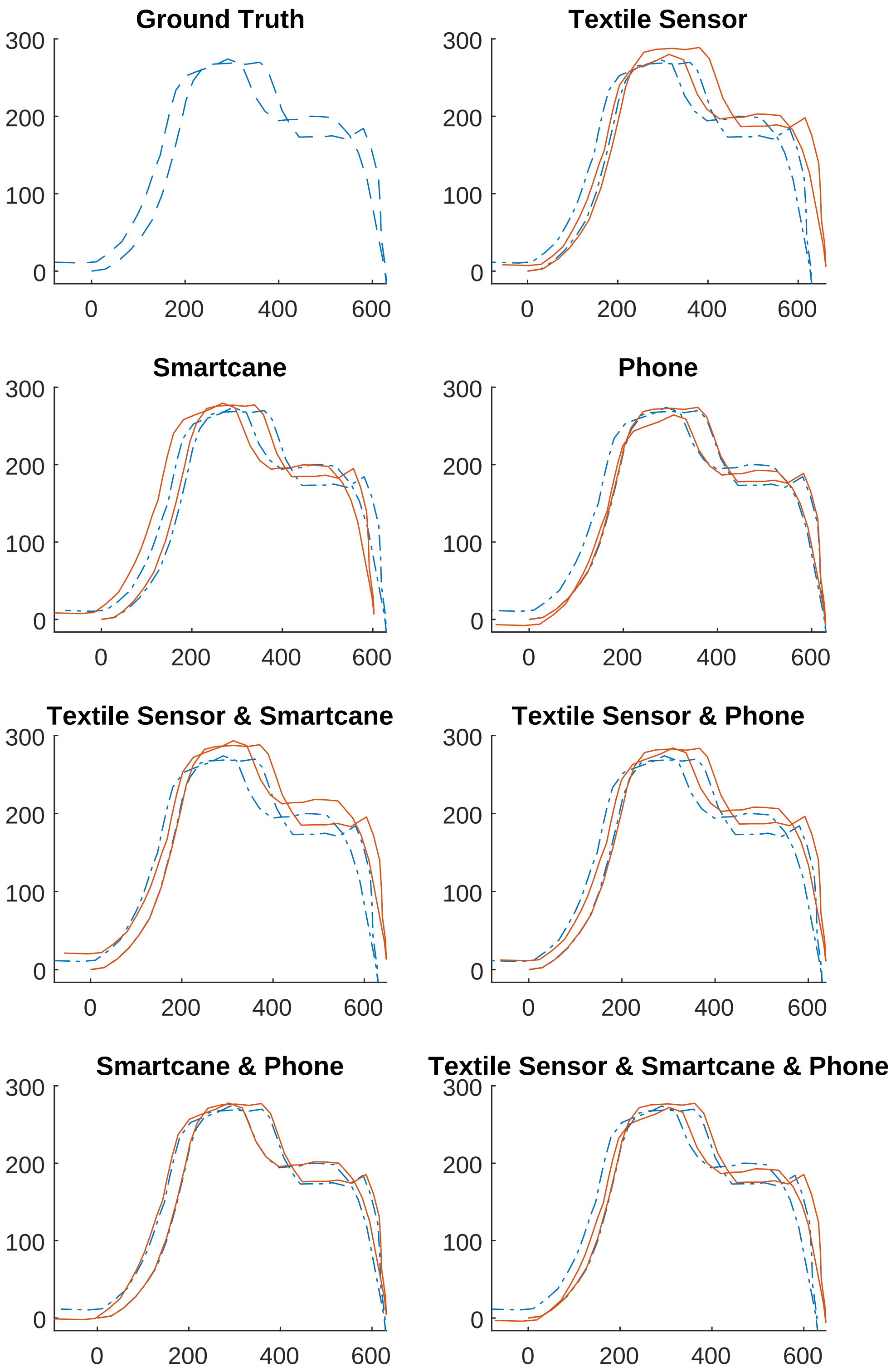

Figure 5.6 : Estimated trajectory of different sensor fusion variations.

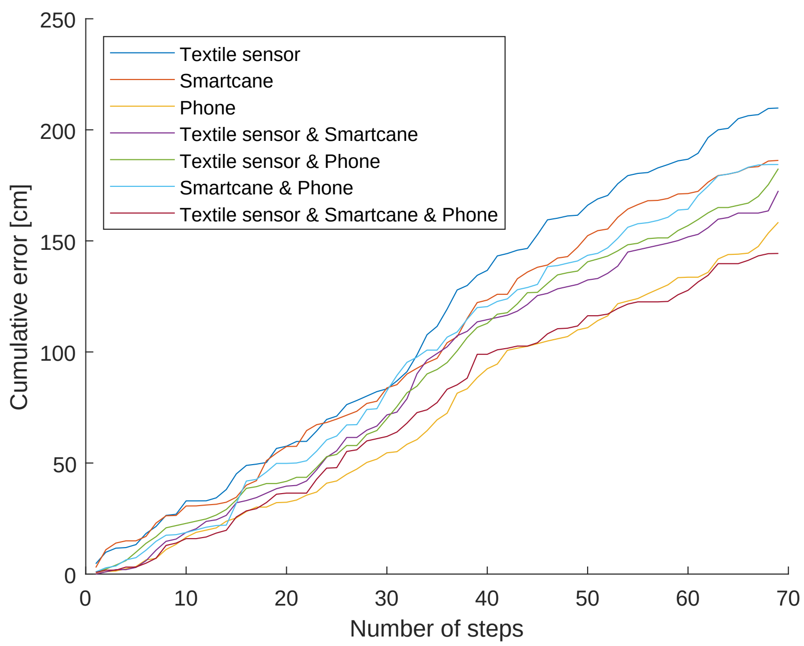

Figure 5.7 : Sum of step length estimation errors for each sensor fusions.

6. CONCLUSION

Navigation applications are widely used to navigate to an unknown location. These applications can find the location of the person on the map by using the signals they receive from GPS satellites. However, these signals suffer losses in closed areas. Therefore, it is not possible to track the position of the person within the building by GPS signals. To solve this problem, there are studies to use WiFi, NFC signals instead of GPS signals in the building. However, in these studies, the location signal generators in the building must be predefined to the system. Therefore, in order for these methods to work, the closed area must be adapted to these technologies. In this thesis, a system that can track the position of the person in the closed area was proposed, regardless of the features of the building.

The system introduced in the thesis was assumed to know the initial position of the person. When the person starts walking, the system estimated the step length and direction of the person. Thus, it calculated how far the person has progressed on the map and updates the person's location. The system consisted of textile-based capacitive sensors, a smartphone and a smart cane developed for the visually impaired. Textile-based capacitive sensors were attached to the person's knees. Thus, the angle values on the knee cap were measured while the person is walking. This angle change gives information about the step. In addition to that, acceleration, gyroscope, and compass data were collected from smart cane and phone. All these data were used for step length estimation with machine learning models. Firstly, sensor data from different subjects were collected on a track with different stride lengths. After the collected data was passed through signal processing methods, the steps taken were separated from the signals according to the start and finish points of the step with the help of an algorithm developed in the thesis. The features that provide statistical information about the step are extracted from the step signals. These features were used for the training of regression models. The trained regression models were used in an Android application developed within the scope of the thesis. In this application, the map was shown with the help of the Google Maps Library. The initial position of the person was marked on the map. When the person starts walking, the regression model estimated the step lengths taken by the person. After the direction of the person is taken from the compass sensor, the current position of the person according to the steps taken was shown on the map. Thus, the instant location of the person was shown on the map even when the location systems such as GPS are off on the person's phone.

In the proposed system, the initial position of the person must be known. In future works, the initial position can be obtained by using predefined location indicator Radio Frequency Identification (RFID) or beacon tags in the building. Orientation of the person is obtained by sensors of the mobile phone. This causes a cumulative orientation shift. To solve this problem in future studies, sensor fusion (mobile phone and smartcane) can also be applied in orientation. In future works, The drifting problem in step length estimation can be decreased by using loop closure and SLAM algorithms.

It is aimed to compare performance with other motion tracking systems and SLAM algorithms in the future. Integrating developed regression models into the navigation systems used today, the shortest way to go from one place to another in closed areas can be obtained. Since the target audience of this study are individuals with visual impairments, the methods developed by this thesis can be used in applications and systems developed for the visually impaired.

REFERENCES

[1] **Kuyucu, C.**, **Ayvaz, U.**, **Ozlem, K.**, **Atalay, A.**, **Atalay, O. and Ince, G.** (2019). Comparative Assessment of Knee Motion Monitoring Technologies, *International Conference on Computer Science and Engineering*, 155–160.

[2] **Rajaganapathy, S.**, **Aravind, B.**, **Keerthana, B. and Sivagami, M.** (2015). Conversation of Sign Language to Speech with Human Gestures, *Procedia Computer Science*, *50*.

[3] **Karakanta, A.**, **Negri, M. and Turchi, M.**, (2020), MuST-Cinema: a Speech-to-Subtitles corpus.

[4] **WHO**, `URL:https://www.who.int/en/news-room/fact-sheets/detail/blindness-and-visual-impairment`, [accessed: 2020-06-01].

[5] **Shahid, S.**, **Nandy, A.**, **Mondal, S.**, **Ahamad, M.**, **Chakraborty, P. and Nandi, G.** (2012). A study on human gait analysis, *Proceedings of the Second International Conference on Computational Science, Engineering and Information Technology*, 358–364.

[6] **Chalmers, E.**, **Le, J.**, **Dulai, S.**, **Watt, J.**, **Andersen, J. and Lou, E.** (2013). Inertial sensing algorithms for long-term foot angle monitoring for assessment of idiopathic toe-walking, *Gait posture*, *39*.

[7] **Perry, J. and Burnfield, J.** (2010). *Gait Analysis: Normal and Pathological Function*, Slack Incorporated.

[8] **Pappas, I.**, **Popovic, M.**, **Keller, T.**, **Dietz, V. and Morari, M.** (2001). A Reliable Gait Phase Detection System, *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, *9*, 113 – 125.

[9] **Jirawimut, R.**, **Ptasinski, P.**, **Garaj, V.**, **Cecelja, F. and Balachandran, W.** (2001). A Method for Dead Reckoning Parameter Correction in Pedestrian Navigation System, volume 52, pp.1554 – 1558 vol.3.

[10] **Salvi, D.**, **Velardo, C.**, **Brynes, J. and Tarassenko, L.** (2018). An Optimised Algorithm for Accurate Steps Counting From Smart-Phone Accelerometry, volume2018, pp.4423–4427.

[11] **Ying, H.**, **Silex, C.**, **Schnitzer, A.**, **Leonhardt, S. and Schiek, M.**, (2007). Automatic Step Detection in the Accelerometer Signal, volume 13, pp.80–85.

[12] **Jayalath, S.**, **Abhayasinghe, N. and Murray, I.** (2013). A Gyroscope based Accurate Pedometer Algorithm.

[13] **Yang, X. and Huang, B.** (2015). An accurate step detection algorithm using unconstrained smartphones, *Chinese Control and Decision Conference*, pp.5682–5687.

[14] **Rozyyev, A.**, **Hasbullah, H. and Subhan, F.** (2011). Indoor Child Tracking in Wireless Sensor Network using Fuzzy Logic Technique, *Research Journal of Information Technology*, *3*, 81–92.

[15] **Wang, Z.**, **Yang, Z. and Dong, T.** (2017). A Review of Wearable Technologies for Elderly Care that Can Accurately Track Indoor Position, Recognize Physical Activities and Monitor Vital Signs in Real Time, *Sensors*, *17*, 341.

[16] **Murata, M.**, **Ahmetovic, D.**, **Sato, D.**, **Takagi, H.**, **Kitani, K. and Asakawa, C.** (2018). Smartphone-based Indoor Localization for Blind Navigation across Building Complexes, *IEEE International Conference on Pervasive Computing and Communications*, pp.1–10.

[17] **Simões, W. and Lucena Jr, V.** (2016). Indoor Navigation Assistant for Visually Impaired by Pedestrian Dead Reckoning and Position Estimative of Correction for Patterns Recognition, *IFAC-PapersOnLine*, *49*.

[18] **Moder, T.**, **Hafner, P. and Wieser, M.** (2014). Indoor Positioning for Visually Impaired People Based on Smartphones, *14th International Conference on Computers Helping People with Special Needs*, p.441–444.

[19] **Renaudin, V.**, **Susi, M. and Lachapelle, G.** (2012). Step Length Estimation Using Handheld Inertial Sensors, *Sensors (Basel, Switzerland)*, *12*, 8507–25.

[20] **Ho, N.H.**, **Truong, P. and Jeong, G.M.** (2016). Step-Detection and Adaptive Step-Length Estimation for Pedestrian Dead-Reckoning at Various Walking Speeds Using a Smartphone, *Sensors*, *16*, 1423.

[21] **Alvarez, J.**, **González, R.**, **Alvarez, D.**, **Lopez, A. and Rodríguez-Uría, J.** (2007). Multisensor Approach to Walking Distance Estimation with Foot Inertial Sensing, *29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 5719–5722.

[22] **Shin, S.**, **Park, C.**, **Kim, J.**, **Hong, H. and Lee, J.** (2007). Adaptive Step Length Estimation Algorithm using Low-cost MEMS Inertial Sensors, pp.1 – 5.

[23] **Retscher, G. and Kealy, A.** (2006). Ubiquitous Positioning Technologies for Modern Intelligent Navigation Systems, *Journal of Navigation - J NAVIG*, *59*, 91–103.

[24] **Shaffer, C.A.**, (2017). GPS/GIS, American Cancer Society, pp.1–7.

[25] **Singh, B.**, **Pallai, S. and Rath, S.** (2012). A Survey of Cellular Positioning Techniques in GSM Networks, *National Conference in Mobile Computing*.

[26] **Balbach, O.** (2000). UMTS - Competing Navigation System and Supplemental Communication System to GNSS, *Proceedings of the 13th International Technical Meeting of the Satellite Division of The Institute of Navigation*, 519–527.

[27] **Singh, B.**, **Santosh, K.**, **Sahoo**, **Ranjan, S.**, **Pradhan**, **Jvcce and Mahavidyalaya, B.** (2014). Analysis of Cellular Positioning Techniques in UMTS Networks, *Journal of Telecommunication, Switching Systems and Networks*.

[28] **MacGougan, G.**, **Lachapelle, G.**, **Nayak, R. and Wang, A.** (2020). Overview of GNSS Signal Degradation Phenomena, *International Symposium on Kinematic Systems in Geodesy (KIS2001), Geomatics And Navigation*, 87–100.

[29] **Ozsoy, K.**, **Bozkurt, A. and Tekin, I.** (2013). Indoor positioning based on global positioning system signals, *Microwave and Optical Technology Letters*, *55*.

[30] **Farid, Z.**, **Nordin, R. and Ismail, M.** (2013). Recent Advances in Wireless Indoor Localization Techniques and System, *Journal of Computer Networks and Communications*, *2013*.

[31] **Dudhane, N. and Pitambare, S.** (2015). Location Based and Contextual Services Using Bluetooth Beacons: New Way to Enhance Customer Experience, *Lecture Notes on Information Theory*, *3*.

[32] **Subhan, F.**, **Hasbullah, H.**, **Rozyyev, A. and Bakhsh, S.** (2012). Analysis of Bluetooth signal parameters for indoor positioning systems, volume 2, pp.784–789.

[33] **Hossain, A.M. and Soh, W.S.** (2007). A Comprehensive Study of Bluetooth Signal Parameters for Localization, *IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications*, pp.1 – 5.

[34] **Chon, H.**, **Jun, S.**, **Jung, H. and An, W.** (2004). Using RFID for accurate positioning, *Journal of Global Positioning Systems*, *3*, 32–39.

[35] **Xu, H.**, **Ding, Y.**, **Li, P.**, **Wang, R. and Li, Y.** (2017). An RFID Indoor Positioning Algorithm Based on Bayesian Probability and K-Nearest Neighbor, *Sensors*, *17*, 1806.

[36] **Davídek, D.**, **Klečka, J.**, **Horak, K. and Nováček, P.** (2016). Odometer Module for Mobile Robot with Position Error Estimation, *IFAC-PapersOnLine*, *49*, 346–351.

[37] **Hu, J.**, **Xu, X.**, **Yang, Y. and Yin, M.**, (2019). Simulation and Experiment on Artificial Landmark-Based Monocular Visual Navigation System for Mobile Robot, pp.118–130.

[38] **Basiri, A.**, **Pouria, A. and Winstanley, A.** (2014). The Use of Quick Response (QR) Codes in Landmark-Based Pedestrian Navigation, *International Journal of Navigation and Observation*, *2014*, 1–7.

[39] **Li, Q.**, **Zhu, J.**, **Liu, T.**, **Garibaldi, J.**, **Li, Q. and Qiu, G.** (2017). Visual landmark sequence-based indoor localization, *1st Workshop on Artificial Intelligence and Deep Learning for Geographic Knowledge Discovery*, pp.14–23.

[40] **Atalay, A.**, **Sanchez, V.**, **Atalay, O.**, **Vogt, D.M.**, **Haufe, F.**, **Wood, R.J. and Walsh, C.J.** (2017). Batch Fabrication of Customizable Silicone-Textile Composite Capacitive Strain Sensors for Human Motion Tracking, *Advanced Materials Technologies*, (9), 1700136.

[41] **Ozlem, K.**, **Atalay, O.**, **Atalay, A. and Ince, G.** (2018). Textile Based Sensing System for Lower Limb Motion Monitoring, *International Conference on NeuroRehabilitation*, Springer, pp.395–399.

[42] **Al-Fahoum, A.S. and Al-Fraihat, A.A.** (2014). Methods of EEG signal features extraction using linear analysis in frequency and time-frequency domains, *ISRN neuroscience*, *2014*.

[43] **Choubey, H. and Pandey, A.** (2019). A new feature extraction and classification mechanisms For EEG signal processing, *Multidimensional Systems and Signal Processing*, *30*(4), 1793–1809.

[44] **Gosztolya, G.** (2015). Conflict Intensity Estimation from Speech Using Greedy Forward-Backward Feature Selection, *INTERSPEECH*, pp.1339–1343.

[45] **Mishra, S.**, **Sarkar, U.**, **Taraphder, S.**, **Datta, S.**, **Swain, D.**, **Saikhom, R.**, **Panda, S. and Laishram, M.** (2017). Principal Component Analysis, *International Journal of Livestock Research*, 1.

CURRICULUM VITAE

Name Surname:Hasbi Sevinc

Place and Date of Birth:Balıkesir, 15 January 1992

E-Mail:sevinch@itu.edu.tr

EDUCATION:

- **B.Sc.:** 2015, Istanbul Technical University, Electrical and Electronics Engineering Faculty, Electronics and Communication Engineering Department
- **M.Sc.:** 2020, Istanbul Technical University, Computer and Informatics Engineering Faculty, Department of Computer Engineering

PROFESSIONAL EXPERIENCE AND REWARDS:

- 2016 - Present, Software engineer in Vestel
- October 2017, 2nd Place in AkbankLab FinTech Hackathon.
- September 2017, 1st Place in Albaraka Turk FinTech Hackathon.
- June 2015, 1st Place in Annual Cansat Competition.

PUBLICATIONS, PRESENTATIONS AND PATENTS ON THE THESIS:

- Sevinc H., Ayvaz U., Ozlem K., Elmoughni H., Atalay A., Atalay O., Ince G., 2020. Step Length Estimation Using Sensor Fusion. *IEEE International Conference on Flexible and Printable Sensors and Systems*, August 16-19, 2020 Manchester, UK.