UNIVERSITY OF OULU
FACULTY OF INFORMATION TECHNOLOGY AND ELECTRICAL ENGINEERING

Adnanul Islam

DETECTING HEAD MOVEMENT USING GYROSCOPE DATA COLLECTED VIA IN-EAR WEARABLES

Master's Thesis
Degree Programme in Computer Science and Engineering
June 2021

Islam A. (2021) Detecting Head Movement Using Gyroscope Data Collected via In-Ear Wearables. University of Oulu, Degree Programme in Computer Science and Engineering, 71 p.

ABSTRACT

Head movement is considered as an effective, natural, and simple method to determine the pointing towards an object. Head movement detection technology has significant potentiality in diverse field of applications and studies in this field verify such claim. The application includes fields like users interaction with computers, controlling many devices externally, power wheelchair operation, detecting drivers' drowsiness while they drive, video surveillance system, and many more. Due to the diversity in application, the method of detecting head movement is also wide-ranging. A number of approaches such as acoustic-based, video-based, computer-vision based, inertial sensor data based head movement detection methods have been introduced by researchers over the years. In order to generate inertial sensor data, various types of wearables are available for example wrist band, smart watch, head-mounted device, and so on.

For this thesis, eSense - a representative earable device - that has built-in inertial sensor to generate gyroscope data is employed. This eSense device is a True Wireless Stereo (TWS) earbud. It is augmented with some key equipment such as a 6-axis inertial motion unit, a microphone, and dual mode Bluetooth (Bluetooth Classic and Bluetooth Low Energy). Features are extracted from gyroscope data collected via eSense device. Subsequently, four machine learning models - Random Forest (RF), Support Vector Machine (SVM), Naïve Bayes, and Perceptron - are applied aiming to detect head movement. The performance of these models is evaluated by four different evaluation metrics such as Accuracy, Precision, Recall, and F1 score. Result shows that machine learning models that have been applied in this thesis are able to detect head movement. Comparing the performance of all these machine learning models, Random Forest performs better than others, it is able to detect head movement with approximately 77% accuracy. The accuracy rate of other three models such as Support Vector Machine, Naïve Bayes, and Perceptron is close to each other, where these models detect head movement with about 42%, 40%, and 39% accuracy, respectively. Besides, the result of other evaluation metrics like Precision, Recall, and F1 score verifies that using these machine learning models, different head direction such as left, right, or straight can be detected.

Keywords: Head Movement Detection, Earables, Gyroscope Data, Machine Learning

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

IoT	Internet of Things
ML	Machine Learning
LGO	Localized Gradient Orientation
SIFT	Scale-Invariant Feature Transform
SVM	Support Vector Machine
RF	Random Forest
IR	Infrared
LED	Light Emitting Diode
LK	Lucas-Kanade
TWS	True Wireless Stereo
CRP	Cardiopulmonary Resuscitation
ECG	Electrocardiogram
CVD	Cardiovascular Diseases
PPG	Photoplethysmography
GSR	Galvanic Skin Response
EEG	Electroencephalography
ESN	Echo State Network
CNN	Convolutional Neural Network
KNN	K-Nearest Neighbor
LDA	Linear Discriminant Analysis
HEMOSCS	Head Movement Controller System
HMD	Head-Mounted Display
ACC	Angel-Based Acceleromete
HAR	Human Activity Recognition
EPW	Electric-Powered Wheelchair
HMI	Human Machine Interface
IW	Intelligent Wheelchair
PDA	Personal Digital Assistant
HGI	Head Gesture-Based Interface
HCI	Human-Computer Interaction
IMU	Inertial Measurement Unit
IQR	Interquartile Range
RMS	Root Mean Square
MMH	Maximum Marginal Hyperplane
MAP	Maximum A Posteriori

1. INTRODUCTION

The things embedded with sensors, electronics, and software connected to internet focusing to collect and exchange data without the involvement of human is defined as Internet of Things (IoT). Connection among any media and anything regardless time and place is achievable now due to enhancement in IoT technology. It is predicted that number of interconnected IoT devices would be exponentially increasing and it would reach up to more than 34 billion devices until 2021 [1].

Perwej et al. [1] also mention that any available technology does not perform with its maximum capability, however IoT is a notable technology that helps other technologies to reach to its full capacity. This technology has huge number of applications covering almost every field of our life. IoT is applicable to make our home smart such as autonomous heating system, smart alarm locks, powerful home security system, and so on. Furthermore, IoT has great impact on waste management by making seamless integration among light, heat, and air cooler. Another vital field where IoT can bring drastic improvement is health care domain. Through utilization of IoT devices, health care services can overcome the limitations in resources like financial and personnel [2]. This technology can assist specialists to reach their patients living far away and can help in dealing with communities of aging populations. In addition, IoT has vast applications opportunities in urban environment which is known as smart city applications of IoT. It includes public services such as transportation, effective water supply, smart traffic lighting, garbage management, and many more. Autonomous car is another domain of IoT application that has drawn high volume of attention from researchers. The research in this sector is exponentially growing.

Huge number of researches have been carried out over the time aiming to make IoT technology more realistic, meaning how IoT can be applied to enhance human capability in various real-life situation. To achieve this objective, head movement detection is considered as one of the significant method. Besides, head movement detection technology has diverse field of applications opportunities such as in assistive technology, teleconferencing and virtual reality [3].

To determine the pointing towards an object, direction of face in terms of communication and interaction, head movement detection is recognized as an effective, natural, and simple method. Out of vast number of applications, one of the key purposes of head movement detection and tracking is allowing users to interact with computers. Additionally, this method provides the opportunity of controlling many devices by mapping the position of the head into control signals [3].

Furthermore, head movement detection technology has been potentially using in health care domain. For example, a wheelchair introduced by Sasou [4] that can be operated using acoustic-based head movement system. User can start operating the wheelchair making sound by breathing and direction of head is employed to set the direction of wheelchair. Similarly, hands-free control of a power wheelchair using the technology of head movement detection was presented by King et al. [5].

The technology of head movement detection has been applied for other purposes as well such as in video surveillance and car assistant system. Nowadays, video surveillance is a core part of everyday life. In order to ensure the security for different purposes, video surveillance system plays significant role. Xie et al. [6] introduced a video-based system using one surveillance camera to capture the face or the head.

Subsequently, the head movement detection system is employed to identify the person of the video. Additionally, for driver car assistant system technology, recognizing the driver attention is exceedingly important. Drivers' head position can assist to identify their attention. By doing that the distraction of driver can be prevented which is significantly beneficial to reduce the number of road accidents. For instance, to identify drivers' attention, Liu et al. [7] employed head posture detection method. Besides, Lee et al. [8] represented a system where head movement is used to realize drivers' drowsiness.

Due to diversity in application of head movement detection technology, the method of detecting head movement is also wide-ranging. The vital aim of increased size of research in this field is to deliver real-time head movement detecting and tracking technology. Therefore, vast number of attempts are carried out to develop method that can identify the movement of head in real-time. Using sensors such as gyroscopes and accelerometer to gain information about head movement is one of the widely applied methods. Nguyen et al. [9] proposed a method that can detect head movement of users by investigating the data gathered from a dual-axis accelerometer and pattern recognition procedure. Applying this method, the head movement can be classified to one of four gestures where an optimized version of Neural Network – a trained Bayesian Neural Network – is used.

Another approach presented by Manogna et al. [10] where on the user's forehead an accelerometer device was attached. This accelerometer device is able to sense the tilt generated from the user's head movement. Then, this tilt is used to produce analogue voltage value. Using this voltage value, control signals can be created to identify the head movements. The study of Kim et al. [11] introduced a head tracking system that can detect head movement using gyroscope sensor data. The data about angular velocity collected via gyroscope sensor is converted to angles by applying an integral operation. In this method, to identity the head posture, relative coordinates are preferred to apply than absolute coordinates.

Additionally, computer vision-based is another preferred head movement detection method. Murphy-Chutorian and Trivedi [12] used image processing and pattern recognition technique in order to develop a static head movement detection algorithm as well as a visual 3-D tracking algorithm. Using these two algorithms real-time technology was developed to identify the position and orientation of user's head. This technology works in three different modules to locate the head position. It detects the preliminary posture of the head and then track the head position and direction in six degrees of freedom. Haar-wavelet Adaboost cascades is used in this head movement detection module. As inputs the preliminary head pose detection section applies support vector machine (SVR) and localized gradient orientation (LGO) histograms. In the tracking module, an appearance-based particle filter technology is employed to pinpoint the 3-D movement of the head. A wonderful driver awareness monitoring system based on head movement estimation was created applying these algorithms.

Moreover, a video-based technique was proposed by Liu et al. [7] for detecting head pose and utilized to create attention recognition technology for drivers. Using this system, the general posture between adjoining views in subsequent video frames can be projected. In order to coordinate corresponding feature points of two contiguous perspective, Scale-Invariant Feature Transform (SIFT) descriptors are utilized. Then two-view geometry is used to find the relative pose angle. This system in general is applicable for image processing area. Additionally, utilizing this mathematical solution, x, y, and z coordinates of the head location can be identified.

Another approach was presented by Kupetz et al. [13]. Make use of an IR camera and IR LEDs, a head movement tracking system was developed by them. It tracks 2x2 infrared LED array which is connected to the rear of the head. A light tracking video analysis method is applied to process the LED motion. According to this method, each frame is sectioned into regions of interest and the development of key element points is followed between frames [14]. In order to supply power for LEDs, the system requires either wire connection or batteries. A power wheelchair controlling feature was developed using this technology.

Jian-zheng and Zheng [15] introduced another idea of detecting head movements where they used mathematical method using image processing techniques. They believe feature points such as nostrils are closely positioned with head, therefore the head movements can be identified by tracking the feature points direction. Aiming to obtain the pattern of positions of head, authors have used Lucas-Kanade (LK) algorithm. To identify the direction of head, they have also trained GentleBoost classifiers to employ the coordinates of the nostrils in a video frame.

Furthermore, Berjón et al. [16] offered combination of alternative Human-Computer Interfaces system that include head movements, voice recognition and mobile devices. An RGB camera and image processing technique were applied to locate head position. In this technique combinedly Haar-like features and an optical flow algorithm are used. The job of Haar-like features is to detect the direction of the face, whereas optical flow algorithm is responsible for detecting the changes of the face position happened within the image.

Acoustic-Signal-Based Method is another approach that has been considered to estimate the head movement. For example, an acoustic-based method for detecting head direction is introduced by Sasou [4]. Author employs a microphone array aiming to detect the source of the sound in this method. Subsequently, in each head direction, the sounds produced by user including its localized positions are distributed around specific areas that is possible to be identified. Hence, by defining the unique areas between the boundaries, it is possible to detect the head movement based on the corresponding areas of generated sounds. This method was applied to develop a power wheelchair control system.

Following table 1 depicts summarization of different head movement detection methods carried out by various researchers:

Table 1. Summarization of different head movement detection method

Authors	**Head Movement Detection Method**
Sasou [4]	Acoustic-based system
Xie et al. [6]	Video-based system
Liu et al. [7]	Head posture based system
Nguyen et al. [9]	Apply data gathered from a dual-axis accelerometer
Manogna et al. [10]	Use voltage value to create signals to identify the head movements
Kim et al. [11]	Use gyroscope sensor data
Murphy-Chutorian and Trivedi [12]	Computer vision-based
Kupetz et al. [13]	Use an IR camera and IR LEDs
Jian-zheng and Zheng [15]	Apply Mathematical method using image processing technique

It is addressed that head movement detection technology has been applied in various field, and vast amount of attempts are already been taken into account to develop real-time head movement identification system, however the challenges for this field still remain high as it requires high computational hardware [3]. A microcontroller - low computational hardware - is not always capable to execute head movement detection algorithms. Therefore, the time required for CPU to process and analyze head movements data needs to be improved. Moreover, developing an appropriate method for head movement detection is also a crucial challenge as study shows the success rate of different methods differ considerably. For example, the study in [3] shows, Computer vision based head tracking system done by Zhao et al. [17] has detection accuracy of 92.86%, on the other hand, accelarometer and gyroscope sensor-based head movement detection method created by [5] and [9] has detection accuracy of 99.05% and 93.75%, respectively.

Considering the potentiality and challenges of head movement detection technology, in this thesis, the aim is set to detect head movement applying machine learning models using gyroscope data. The performance of these ML models can be evaluated by measuring the accuracy result of models. Besides, this thesis also set a target to assess whether the detection of head movement is able to enhance human capability in given scenario such as driving or biking. This can be evaluated comparing the result and findings of this thesis against existing literature. To overcome the challenges mentioned earlier, eSense device [1] which has high computational functionality to generate gyroscope data is used in this thesis. The contribution of this thesis in this field would be detecting individual's head direction - left, right, or straight - using

[1]https://www.esense.io/share/eSense-User-Documentation.pdf

gyroscope data generated through eSense device. Thus, the real-time applications for different real-life situations can be created utilizing this detection model.

Gyroscope data mainly measure angular velocity. The angular velocity is counted in degrees per second or revolutions per second. In simple words, angular velocity is the measurement of speed of rotation. eSense is a multi-sensory earable platform that is chiefly used for analytical research of personal-scale behavioral [18]. This eSense device is a True Wireless Stereo (TWS) earbud. It is augmented with some key equipment such as a 6-axis inertial motion unit, a microphone, and dual-mode Bluetooth (Bluetooth Classic and Bluetooth Low Energy) [2]. This device has two earbuds, thus it generates more fine-grained data points that enhance the precision of the estimation.

Data collected via eSense platform is preprocessed in order to implement feature extraction. Subsequently, four Machine Learning (ML) models; Random Forest (RF), Support Vector Machine (SVM), Naïve Bayes, and Perceptron are applied to detect the direction of face. All machine learning models are applied using determined features values. Then, the accuracy, precision, recall and f1 score of models are measured to evaluate the performances. The importance of different features has also been analyzed in this thesis.

The Figure 1 given below illustrates all the steps that have been applied in this thesis aiming to identify users head movement using different machine learning models:

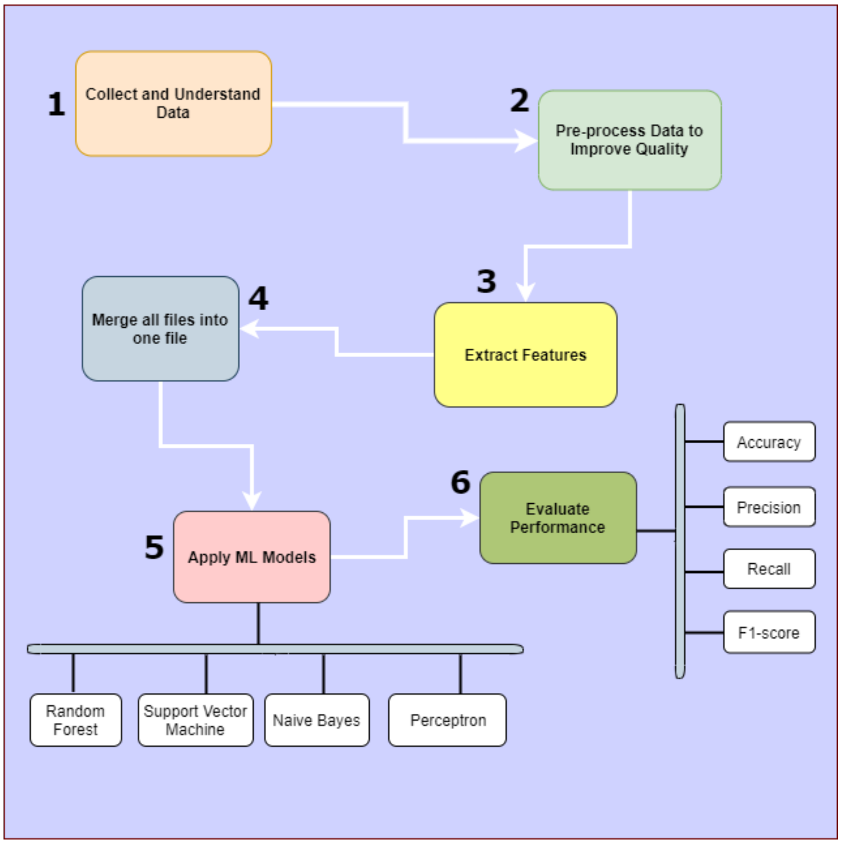


Figure 1. Process pipeline of this thesis

[2]https://www.esense.io/share/eSense-User-Documentation.pdf

This thesis focuses on following research questions:

(RQ1) How accurately the head movement can be detected using gyroscope data collected through in-ear wearables?

(RQ2) Determine whether the head movement detection is able to enhance the human capability in given scenario such as biking or driving?

In order to represent the thesis in a structured way, the following sections are performed. Section 2 talks about related work covering wearable technology including earables, various approaches of head movement detection, application of gyroscope sensors in head movement detection as well as other field of science. The dataset description and method of collecting data is illustrated in section 3. Section 4 discusses about theoretical background of different applied machine learning models including the evaluation methodology. In section 5, the implementation process is explained. The result and the analysis of it are described in section 6. Discussion including findings, limitation, challenges, and future work is represented in section 7. Finally, the conclusion is provided in section 8. In section 9, all the references are listed.

2. RELATED WORK

Earables – in-ear wearables – which is employed in this thesis to collect data has great opportunities and mostly unexploited prospectives. Earables have the potentiality to be used as sensors and actuators [19]. Earables possess an extremely good vantage in terms of generating accelerometer and gyroscope data in different circumstances like while exercising, performing daily life activities, as well as while biking or driving. In this study [20], it is represented that how a system that relies only on accelerometer and gyroscope data can still deliver a meaningful insight regarding where a person is facing. The key objective of this thesis is to detect head direction utilizing gyroscope sensor data collected via eSense, thus related work section mainly focuses on discussion about various applications of different types of wearable sensors. The application of earables is also covered, highlighting the difference between earable and other wearable devices. Besides, in this section, studies related to several head movement detection methods and their purposes are described as well. Additionally, various applications of gyroscope sensor is discussed too in this section.

2.1. Wearable Technology

Wearable signifies as all forms of computational or sensory electronic devices. These devices typically can be worn with clothing or on the body. In other words, any computer device that user carries with them aiming to help them for any specific reason could conceivably be called a wearable [21]. In today's digital world, the term wearable no longer means as clothing of elegant dresses. Instead, it brings up the image of different accessories such as a tiny sensor in cyclist's helmet, smart cloth that a runner wears to track and monitor steps, smart watch, smart wrist band, head-mounted displays for gamers, and so on [22].

Various state-of-the-art smart wearables are listed below [23]:

- Xiaomi Amazfit sport watch
- Fitbit Flex wristband
- Google Glass
- Heddoko smart clothing
- LifeBEAM Smart Helmet / Sports Cap
- Solepower SmartBoot

Fundamentally, the following basic functions can be performed by wearables [22]:

- Sense
- Process (Analyze)
- Store

- Transmit
- Apply (Utilize)

As wearable can sense, collect, and upload data continuously regardless time and place, therefore it provides the opportunity of improving daily life which is not always easily possible with smartphone alone. This technology can serve other purposes as well such as indoor localization and navigation [24], [25], sports analytics [26], and so on. It is predicted that wearable technology market value would reach up to 57,653 million US dollar by 2022, it is almost 3 times higher than the 2016 market value (19,633 million US dollar) [27]. Due to increase demand of this technology, vast amount of research is in progress aiming to develop novel wearable devices, as well as focusing on how utilization of wearables in ubiquitous computing can be enhanced. For example, a device namely Gait-Watch [28], a context-aware user authentication system, is developed based on the gait pattern recognition from a smart watch. A novel sparse fusion method is proposed in this study to improve the accuracy. In order to validate the performance, the entire system is implemented on a Samsung Gear Live smart watch. It is evident that Gait-Watch is able to authenticate user with 95.2% true positive rate.

Moreover, a smart watch application was designed by Gruenerbl et al. [29] that can help inexperienced people with performing Cardiopulmonary Resuscitation (CPR). This application use acceleration data produced through smart watch and can guide its user providing feedback on frequency and compression depth. Xu et al. [30] represented another technology that can detect finger, hand, and arm gestures accurately. This technology monitors the movements of tendons passes through wrist using an accelerometer and a gyroscope attached to a smart watch. Authors developed a classifier that is able to identify the characters, written on a surface by index finger, with 95% accuracy.

Wrist bands also have drawn attention from researchers due to its advantage compare to other devices such as it can be worn while sleep or take bath. For example, authors in [31] proposed a new hardware called bio-impedance sensor for wrist band. The intend function of this technology is to deliver accurate implicit user identification. It is claimed that this system can authenticate user with 98% accuracy, the robustness, however, has not been verified as other factors such as wrist orientation, skin temperature, and diet can influence the bio-impedance. The study of Parate et al. [32] proposed a machine learning algorithm that can recognize smoking gestures. In order to do so, a wrist-worn 9-axis IMU featured with an accelerometer, a gyroscope and a magnetometer is employed. The results indicate smoking gestures can be detected with 91% precision and 81% recall rate. In addition, a research that was financed by the EU FPS IST program [33] called AMON or the advanced care and alert portable telemedical monitor. The outcome of this research was the development of a wrist-worn device that can determine blood pressure, skin temperature, blood oxygen saturation, and a one lead ECG. The device was integrated with a two-axis accelerometer to identify correlation between user activity and measured vital signs.

Research also focuses on other various applications of wearables. For instance, the MyHeart project [34] that was supported by the European Commission as well as 33 partners from 10 different countries were involved including industrial partners like Nokia, Vodafone, Philips, and Medtronic, developed smart clothes aiming to fight against cardiovascular diseases (CVD) by prevention and early diagnosis. In this technology, sensing modules were integrated with garments or it simply embedded on the piece of clothing [35], [36]. Pandian et al. [37] proposed a Smart Vest technology which is a wearable physiological monitoring system made of a vest. This technology uses various sensors incorporated on the garment's fabrics. It can collect several biosignals simultaneously in a manner of noninvasive and unobtrusive. Using this system ECG, photoplethysmography (PPG), heart rate, blood pressure, body temperature, and galvanic skin response (GSR) can be calculated.

2.2. Earables

Earables are relatively a new concept. This technology has great potentiality, the research, however, in this field is still not remarkable [19]. Mostly it has been using in health care domain. LeBoeuf et al. [38] introduced prototype of an optomechanical sensor that can sense blood flow and can assist to estimate consumption of oxygen during daily activities. Additionally, a study carried out by Bedri et al. [39] and Amft et al. [40] where they presented a combination of IMUs and microphones to identify and classify eating activities. Initially authors aimed to detect in-the-wild chewing activities, latter they focused to determine classification of four kind of foods by analyzing eating activities.

Another prospective field of application for in-ear wearable is emotion monitoring. Emotional state is a crucial factor that can significantly impact on mental state, thus can greatly influence decision making [41]. The study of Athavipach et al. [42] represented an in-ear EEG wearable device that is low cost, single channel, and dry contact. This device is appropriate for non-intrusive monitoring. Authors applied machine learning models for emotion classification. Applying their valence and arousal emotion model, the device is able to classify basic emotion. The models identified emotions with 71.07% accuracy for valence, 72.89% accuracy for arousal, and 53.72% for all emotions.

Moreover, in brain-computer interface domain, eye blinks or different facial expression in EEG signal created by muscle activity are examined aiming to control external devices. To illustrate, an in-ear headset using hacked NuroSky EEG sensor developed by Matthies et al. [43]. This prototype can be used for explicit control of the functionalities of a smartphone employing eye winking and ear wiggling. In addition, Matthies et al. [44] presented a device in 2017 where they developed a foam earplug placing multi electrodes on it. This device is able to detect 25 types of facial expression as well as head gestures with four different sensing technologies. With accuracy of 90% and above 50%, five gestures and fourteen gestures could be detected, respectively. It is evident that the prototype is also robust under given practical scenario such as walking.

Considering the importance of maintaining attention while carrying out crucial daily life tasks that need high level of safety and efficiency, number of different brain activity measurement devices like electroencephalography (EEG) have been utilized to observe the attention state in individual. In [45], Jeong et al. designed and built an in-ear EEG electrodes fitted in both the left and right ear canal of the user. Based on in-ear EEGs, an echo state network (ESN) and a machine learning algorithm are applied to differentiate the attention states of the subjects. The result of the study shows that with optimal network parameters the ESN method can discriminate between attentive and resting states with maximum average of about 81.16% accuracy. A research carried out in [46] introduced an earpiece that is able to robustly measure the brain, cardiac and respiratory functions. For future health system it is required to assess and track neural and physiological functions of a user throughout long period of time. Consequently, such technology needs to be developed that can be fitted comfortably on body as well as can perform desire functionality. Addressing these challenges, authors, in this research [46], have developed the earpiece that benefits from its considerably stable position of the ear canal with respect to vital organs.

Furthermore, earables can be useful device to track drivers' head movement in order to make sure drivers are awake, alert, attentive, as well as looking in the right direction. The hands-free and immediate interaction functionality of earable devices can be leveraged as a contextual communication interface. This interface can assist to deliver drivers information related to routes, more importantly it can prevent drivers from reading text while they drive. Additionally, earables can sense users' physical, social, and environmental contexts, thus this device can offer personalized services more persuasively. Head and mouth related activities such as eating, speaking, drinking, shaking, and nodding can also be effectively tracked by appropriate utilization of earables. This technology can be successful to monitor minute head and neck movements, therefore can be applicable for different clinical medicine application concerning neck and head injury [18].

eSense platform, an in-ear multisensory stereo device, which has been applied in this thesis to collect data aiming to identify head movement, has also been used in other studies for various examination. Such as in [18], authors have applied this earable device for personal scale behavior analytics. They have claimed that the knowledge of a wide range of human activities in a nonintrusive manner can be accelerated using this device. Lee et al. [47] introduced a method to collect inertial signals through an earable positioned in the ear canal. It is presented as a new compelling sensing modality which is capable of recognizing two vital facial expressions - smile and frown. Utilizing the principle of Facial Action Coding Systems, authors then illustrated facial muscle deformation triggered by a set of temporal micro-expression can be captured by inertial measurement unit of this specific earable device. Based on these understandings, the learning models – shallow models with statistical features, hidden Markov Model, and deep neural network - are presented capable of automatically detecting smile and frown from inertial signals. The outcome of this study shows that in controlled non-conversational situations, this system can distinguish smile and frown with high accuracy - F1 score: 0.85.

In this digital era, in our everyday computing experiences, conversational agents are becoming exceedingly important as it offers various purposeful information and utility services. Considering the fruitfulness of this service, however these agents are not entirely capable of realizing their users' situational or emotional states, thus incompetent in terms of adapting their tone and interaction style based on context. To this end, in [48] authors proposed a first-of-its-style situation-aware conversational agent built on eSense platform that is capable of adjusting its conversation style, tone, volume based on its users' emotional, situational, social, and activity perspective. This device recognizes users' context through speech prosody, ambient sound, and motion signatures. To be specific, the system consists of four key components such as perception builder, conversation builder, affect adapter and text-to-speech builder. Users are required to wear the eSense earables to make conversation with the agent and a Raspberry Pi Zero is used to do the computation process.

Exercise and physical activity can benefit people in various way such as keeping fit mentally and physically. People who perform daily exercise are healthier and have better mood. This habit can lessen the risk of having several chronic diseases such as cardiovascular disease, diabetes, cancer, hypertension, obesity, and depression. A research carried out by Ishii et al. [49] introduced ExerSense that can segment, classify, and count several physical exercises in real-time on the basis of correlation method. In this study, authors have employed four different wearable devices including eSense earable to collect acceleration data for five types of regular exercises. In terms of correctly extracting segments, eSense perform with 84% recall score. To validate the classification method, the result shows eSense gain 76% accuracy. In both cases, eSense perform the worst compare to other devices positioned on the chest, upper arm, smartwatch mounted on the wrist.

Additionally, using eSense device, authors in [50] proposed an activity recognition framework. The accelerometer and gyroscope data using eSense were collected aiming to detect head and mouth related activities, as well as other daily human activities. To achieve the objectives of this research, authors developed a smartphone application for colleting data via eSense. They examined several statistical features to understand head and mouth related activities for example head nodding, shaking, eating and speaking, and other daily activities like stay, walk, and speaking while walking. Authors applied different types of machine learning methods such as Convolutional Neural Network (CNN), Random Forest (RF), Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Linear Discriminant Analysis (LDA) to classify activities. The result of the research shows the performance of the machine learning models were satisfactory. Authors achieved accuracy of 80.45% by LDA, 93.34% by SVM, 91.92% by RF, 91.64% by KNN, and 93.76% by CNN. Both accelerometer and gyroscope sensor data together were exploited in this study.

The technology of identification of human traits and emotional states can be applied across various sectors including healthcare, human-robot interactions, humanoid research, and many more. Individual's traits and emotional states significantly influence person's activity and behavior, and due to these reasons activity and behavior varies from person to person. Thus, it is imperative that if identification of traits and emotional states is achieved, the individual's activity and behavior would possibly get predicted and generated. To this end, in this research [51], authors explored a common human activity – head movement - aiming to understand its relationship with human traits and emotional states. To achieve this objective, researchers analyzed translational and rotational human head movement data. Machine learning methods were applied to link the human traits with subsequent head movement data. Authors used eSense device which has built in accelerometer and gyroscope sensors to collect data about head movements. The result proves the existence of underlying relationship between spontaneous head movement and corresponding traits and emotional states. Fourteen types of human traits and five kind of emotional states were examined in this study.

Using earables have some unique opportunities as well as advantages compare to other wearable devices [52].

- As earables is placed in the ear, therefore it is able to monitor movements related to head and mouth along with whole-body movements in a non-invasive way. Due to this advantage, earables reveal the opportunity for many novel applications such as in personal healthcare, dietary monitoring, and attention management areas.
- Earables are comfortable and discreet that enables users having immediate and hands-free access to information in a manner of privacy-preserving and socially acceptable way.
- The freedom of movement and hands-free interaction offered by earables play crucial role in minimizing situational disability and fragmentation of attention.
- Earable devices can be worn for long hours without having any impact on primary motor as well as cognitive activities.

Wearables including earables are comprised with a variety of sensors, thus the data collected via sensors differ as well. A study carried out in [53] to examine the varieties in sensors supplied in wearbales. This study explored 140 different wearables to identify equipped sensors. The result of the study Figure 2 shows that Heart Rate and movement related sensors such as accelerometer, gyroscope, GPS and Compass are the most common sensors supplied in wearables:

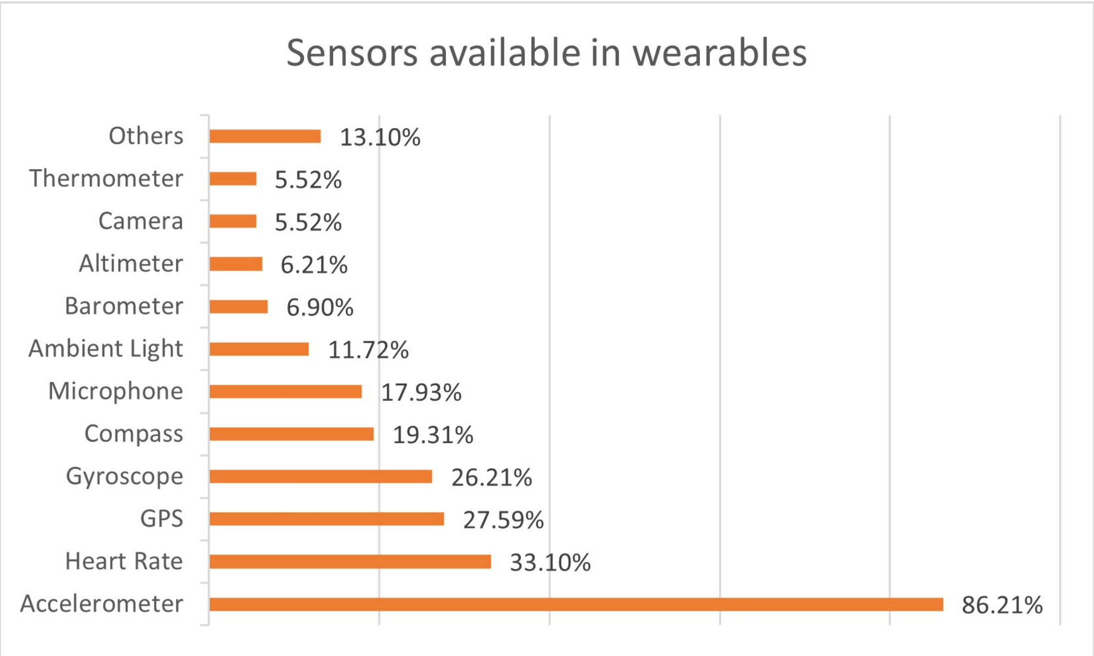


Figure 2. Sensors in wearables

2.3. Head Movement Detection

Head movement tracking is very well explored area of research and the application opportunity of this technology is huge. A number of approaches are examined to detect head movement, and subsequently are used to develop various real life applications. One of the most recent research carried out by Shen et al. [54] where authors introduced a magnetometer-centric sensor fusion algorithm called MUSE for location tracking and the problem of 3D orientation also has been highlighted in this paper. The result indicates that the MUSE performs significantly better than other approaches. In another study done by Tolle et al. [55] designed a head movement controller system (HEMOSCS) which is able to control mobile application via head pose movement identification. In this paper, authors presented a new model of head movement detection system that employs 3 degrees of freedom of head positional changes. A Head-mounted Display (HMD) was created to attach a mobile phone on user's head. The method identifies the internal accelerometer and gyroscope sensor data produced inside the mobile phone positioned on user's head. For implementation of the method a real time mobile application was developed, and outcome verify that user could control hands-free application only using specific head pose movement.

Data shows that number of traffic accidents all over the world are exponentially increasing. Drivers with less concentration, tiredness, sleepiness cause serious danger for people lives including their own one. In this regard, technology that can actively track divers' level of vigilance and alert them for any insecure driving condition is extremely required to prevent accident to save more lives. Having the encouragement from this serious real life issue, authors in this paper [56] outlined that detecting drivers' drowsiness can significantly help to reduce the number of road accidents. Drowsiness can be identified using following categories such as identifying physiological characteristics, driver operation, vehicle response, and observing drivers' response. It is evident out of all these methods, sensing physiological characteristics perform most accurately and it can be achieved by analyzing head movements. Authors utilized a single step accelerometer ADXL330 which calculates 3-axis detection to identify head position. It consists of angel-based accelerometer (ACC) input to simulate the head movement accurately.

Human activity recognition (HAR) is another field of research where detecting head movement plays crucial role. Accurate recognition of human activities is an important technology in pervasive computing as it would allow to develop applications for may real-life human-centric problems like eldercare and healthcare [57]. Concerning this issue, a study done in this paper [58] presented a methodology that can identify 12 kind of human actions on either the frontal or the lateral view. The head pose of users were monitored over successive frames of a monocular grayscale image sequence to pinpoint different human actions. Head movement is segmented in each frame, successively the feature vectors are extracted. As per this method, images were captured by a fixed CCD camera work as input sequence and then matched with stored models of actions. To test the performance of this system, authors applied the nearest neighbor classifier.

Electric-powered wheelchair (EPW) is considered as significant factor in terms of improving the quality of life for elder and disabled people. Getting motivated by this particular concern, researchers represented a number of approaches to develop hands free human machine interface (HMI) that can control wheelchairs using shoulder, head and tongue motion, as well as using eye tracking. In this context, Rechy-Ramirez et al. [59] introduced a user-friendly human machine interface to control an electric powered wheelchair hands free. It has two operation modes which functions based on head movements. In Mode 1, only one head movement is needed to provide the command, whereas mode 2 utilizes four types of head movements to operate the wheelchair. In this study, Emotiv EPOC - an EEG device - was employed to develop the HMI in order to collect data of users' head movements. The performance of proposed HMI was compared to traditional joystick controlled electric powered wheelchair in an indoor environment. The result of the experiment shows Mode 2 performs better than Mode 1. Mode 2 can be executed in a mean time of 67.90 seconds for two subjects in a fast reliable way, whereas for Mode 1 even though it requires only one head movement, takes a mean time of 153.20 seconds to start operating. Authors have claimed the proposed HMI can effectively be used for elderly and disabled people in place of traditional joystick-controlled wheelchairs.

Similarly, a study is carried out by Gray et al. [60] to represent a novel method of controlling intelligent wheelchairs (IWs) using visual identification of head movements. Authors developed a head gesture-based interface (HGI) named RoboChair on the of combination of two algorithms - the Adaboost face detection algorithm and the Camshift object tracking algorithm - aiming to achieve face detection, tracking and gesture recognition accurately. The interface is performed in real time face recognition and tracking. The system is considered as extremely helpful for people who are suffering restricted limb movements, would allow them to operate the intelligent wheelchair with head gestures instead using their hands. In addition, considering the problem of spinal cord injury or motor neuron diseases, the project in [61] presented a wheelchair that can be controlled by head movement. Authors have designed and implemented a wireless head movement-based wheelchair using personal digital assistant (PDA) artificial intelligence technique. The whole technology was embedded on Linux operating system.

A vast number of methods have been applied to build powered wheelchairs in order to facilitate people who are suffering unique physical challenges, to ensure they are not losing their personal independence. For example, SENARIO [62], VAHM [63], Rolland [64], SIAMO [65], Wheelesley [66], and omniwheeled platform [67]. Similarly, a solution is offered in this paper [68] where authors developed a microcontroller system to control a standard electric wheelchair using head motion. This system was implemented and was experimentally tested, it consists of a digital system integrated with an accelerometer and a microcontroller and a mechanical actuator. For collecting head motion data, the accelerometer was employed. Subsequently, the data was processed using a novel algorithm created using a microcontroller.

Another motivation for researchers is developing mobile interface in a way that can function according to user situation, rather demanding user to sit and provide command using touch screen. Concerning this matter, a new method introduced by Elleuch et al in [69] for monitoring tablets via head movements and eye-gaze gestures identification. The proposed system is comprised with different modules and each module has different functionality. The front-facing camera of the tablet is used to capture live video which works as basic input. Then, user face is detected through face detection module utilizing video input. Successively, a face monitoring subsystem is applied to track all the movements of face. Consequently, motion generated through head movements can be detected and execute the actions accordingly. Authors have borrowed techniques from P. Viola and M. J. Jones [70] in order to detect face and used Haar classifier for detecting and tracking eyes. The entire system is developed on android based tablet.

Smart mobile phones are equipped with sensors which can be employed to enhance the device functionality. In the field of education, mobile phones hold huge potentiality. Such as it can be used as an alternative method and can deliver accessibility for users in terms of human computer interface solutions. In [71], applying inertial sensors for example accelerometer, gyroscope, and magnetometer in mobile devices, a head movement detection technology has been developed to navigate mobile application. The vital functionality of this method is it changes the function of hand-based input to head-based input. Head-Mounted Displays (HMDs) are combined with mobile devices in this system that is able to provide a low-cost innovation human-computer interaction. The system can detect six types of head poses movement. In this study, authors designed an interactive mobile technology using head poses movements that can enhance the usability of exam application for disabled students.

Head movements technology posses great potentiality in the context of speech. People make head movements when speak, and these movements are not random. These are the structural sign of ongoing discussion and are used to control interaction. Researchers are focusing on head movements in the context of speech for two vital potentiality, such as head movement play crucial role in speech production, and communicative functions. Hadar et al. [72] realized that while people speak the head moves constantly, whereas stillness largely happen during pauses and listening. Authors also determined head movements correlate the speech in a parallel manner, it creates the relationship between manual gestures and speech. In short, speakers generally move their hands and limbs, while listeners have tendency to stay still.

Another interesting motivation for researchers is to understanding how head movements can control interpersonal interaction without even the presence of speech. In article [73], Adam Kendon carried out an experiment to show how a young woman sitting on park bench can control the activities of her loving partner with only by her head and facial movements. Besides, a study regarding the linguistic functionality of the head movements in the context of speech done by McClave [74] . This study identified that while some head movements express propositional content, others convey semantic meanings that is beyond affirmation and negation. Expression of inclusive and intensification is articulated by side-to-side head shaking, while uncertain speeches and lexical repairs are formulated by lateral head movements.

Table 2. Summarization of different head movement detection method and their purposes

Authors	**Method**	**Purpose**
Shen et al. [54]	Used magnetometer-centric sensor fusion algorithm called MUSE	For Tracking location and the 3D orientation of head
Tolle et al. [55]	Applied a head movement controller system (HEMOSCS)	To control mobile application
Varma et al. [56]	Used a single step accelerometer ADXL330	To identify drivers' drowsiness
Madabhushi et al.[58]	Employed a fixed CCD camera to capture images that work as input sequence and then matched with stored models of actions	To identify 12 kind of human actions on either the frontal or the lateral view
Rechy-Ramirez et al. [59]	Developed user-friendly human machine interface using an EEG device named Emotiv EPOC	To control an electric powered wheelchair hands free
Gray et al. [60]	Created a head gesture-based interface (HGI) named RoboChair	For controlling intelligent wheelchairs (IWs)
Craig et al. [61]	Built a personal digital assistant (PDA) using artificial intelligence technique	To control a wheelchair
Pajkanovic et al. [68]	Developed a microcontroller system consists of an accelerometer, a microcontroller and a mechanical actuator	To control a standard electric wheelchair
Elleuch et al. [69]	Used a head movements and eye-gaze gestures identification system	For monitoring mobile and tablets
Fanani et al. [71]	Applied inertial sensors such as accelerometer, gyroscope, and magnetometer	To navigate mobile application

2.4. Gyroscope Sensor

Gyroscope sensor is a device that is capable of measuring and maintaining the orientation and angular velocity of an object. Gyroscope sensors are comparatively more advanced than accelerometers. Gyroscope can measure the tilt and lateral orientation of an object, on the other hand, only linear motion can be measured by accelerometer.

Gyroscope sensors are also known as Angular Rate Sensor or Angular Velocity Sensors. When humans struggle to sense the orientation of an object, these gyroscope sensors are installed in applications. Angular velocity is calculated by measuring the changes in the rotational angle of an object per unit of time, calculated in degrees per second. Along with measuring the angular velocity, gyroscope can measure the motion of an object as well. In order to have more robust and accurate motion sensing, gyroscope sensors are placed with accelerometer sensors combinedly in most consumer electronic products.

There are three kind of angular rate measurements based on direction [3]:

- Yaw – the horizontal rotation on a flat surface and the object is seen form above.
- Pitch – the vertical rotation and the object is seen form front.
- Roll – the horizontal rotation and the object is seen form front.

The Figure 3 below shows a typical look of a gyroscope sensor (image is taken from Google which is publicly available to use for any purpose):

Figure 3. Gyroscope Sensor

Gyroscope sensors are potentially used in versatile applications. One of the such field of application for gyroscope sensor is Human-Computer Interaction (HCI). In HCI, hand based input using mouse, keyboard, and touch screen is commonly applied method. However, this method has some shortcomings particularly for person who has a broken hand, or has stroke problem, or disable person, would not able to use

[3] https://www.elprocus.com/gyroscope-sensor/

his hand, consequently would be difficult to use devices including mobile phones. In this context, HCI approach like using head can be an alternative to operate different devices. With the motivation of such problem, recently Guan et al. [75] proposed sensor-based approach for HCI where sensors like accelerometer and gyroscope are applied as the input. Nintendo Wii Remote is a great example of sensor based HCI approach [76].

In today's world, modern smartphone devices are equipped with sensors like proximity, gyroscope and accelerometer that are capable of measuring motion, orientation, and different environmental conditions [77]. In this study [78], authors discussed a method of gesture detection using inertia data collected through gyroscope and accelerometer sensor. The study reveals that for rotational movement, gyroscope sensor is correctly appropriate. In addition, Priandani et al. [79] carried out a research where they proposed a head movement identification approach on the basis of Android internal gyroscope sensor. This paper illustrated how head movement works as a new method of Human-Computer Interaction on Android smartphone. Nearest-Neighbor classification algorithm was used for head movement recognition. The robustness of the method was verified with a satisfactory result of 95% accuracy and 2.5 ms execution time of identification process. Authors have claimed the proposed method is suitable in various hardware environment for experiencing real-time human-computer interaction only utilizing head movement.

A similar study done by Yunardi et al. [80] aimed to apply head movement gestures to create hands free human-computer interaction for disable people. This paper focused to utilize head movement detection method to control a robotic manipulator in an assisting device. Authors applied combination of two algorithms – the visual and gyroscope sensor – to implement the process of head movement identification with high precision. The outcome of the result indicates the correct head movement can be detected with average 82% accuracy. Furthermore, Tadano et al. [81] done a research to develop a robotic arm aiming to hold and manipulate a laparoscopy. To operate the robotic arm, user's head movement data was employed. Gyroscope sensors were attached to the user's head and body to measure the head movements. An experiment was conducted to justify the effectiveness of proposed approach. Result shows the robotic camera holder is capable to swiftly follow the user's head movement.

In this related work part, other applications of gyroscope sensor data apart from head movement detection has also been explored. To exemplify, in the field of activity recognition, gyroscope sensor is applied in massive number of studies. It is acknowledged accurate classification of activity in smart phones can help to track and analyze daily activities of users, thus can help to build prospective healthcare applications. Having motivation from such concern, Jain et al. [82] introduced a descriptor-based method to differentiate smart phone activity using built-in accelerometer and gyroscope sensor data. Time and frequency domain signals are obtained from the signals collected via accelerometer and gyroscope. Two descriptors – histogram of gradient and centred signature-based Fourier descriptor – are applied in this proposed technique to achieve feature extraction from these signals. Authors investigated feature level fusion and score level fusion for information fusion. The performance of two machine learning models like multiclass support vector machine and k-nearest neighbor are examined for classification. The performance was evaluated by applying the method into two publicly available data sets.

Moreover, in [83], authors developed a novel automated technique to classify human activities utilizing wearable sensors that are commonly interfaced nowadays with most of the smart mobile phones. To identify the activity pattern, K-nearest neighbours algorithm with three nearest neighbours was used. Activities such as sitting, standing, walking, sleeping and jumping were detected applying this approach with 94% accuracy. Muset et al. [84] carried out a project aiming to develop an alternative method to GPS for measuring distance traveled by an individual inside a building. To achieve this goal, authors implemented an algorithm that is capable of measuring distance by calculating the number of steps completed by a person. Algorithm was executed using accelerometer and gyroscope data to compute the angle between legs, subsequently determine the traveled distance.

Furthermore, Bonnet et al. [85] presented a study concentrating to develop a data processing technique that is able to measure instantaneous 3-D position of an inertial measurement unit (IMU). IMU is mainly consists of accelerometer and gyroscope sensors that are capable of measuring three accelerations and three angular velocities, respectively. In this study, the data was collected through accelerometer, gyroscope, and magnetometers, and subsequently was used to obtain result by fusing. The method offered in this study only based on the use of gyroscope data. Authors applied a Weighted Fourier Linear Combiner adaptive filter for a drift-free estimation of 3-D positioning of an IMU. The IMU was placed on the lower trunk of the subject during walking in this experiment. The effectiveness of the method was validated as well.

Considering the fact that fall situations can be very risky for people who are suffering from different health problems particularly for elder people, authors of this paper [86] presented a method that can detect a fall situation and is able to send notification to colleagues in real-time manner. This system shows high accuracy for certain fall situations. A threshold-based fall identification algorithm was developed for this system due to its comparatively easier computational process. The algorithm mainly focused to identify dynamic situations of postures, as well as to detect unintentional falls to lying positions. Consequently, linear acceleration and angular velocity of the user were recorded applying this algorithm, afterwards compared with set of thresholds collected via training data of monitoring subject.

Data provided by World Health Organization shows that all over the world approximately 1.25 million people die because of road accidents. Studies in [87] explain poor road conditions are the crucial reason to cause road accidents. Therefore, developing road condition monitoring system has gained significant importance in recent years. Extensive research in this field can guarantee safety and comfort to travelers, as well as can help to lessen the damage of cars. Concerning this vital issue, authors of this paper [88] proposed a road condition monitoring system developed based on accelerometer, gyroscope, and GPS. The system is a real time Android Application named RoadSense. The proposed system used machine learning methods to deliver information about the quality of road of different zones. To classify the road segment, C4.5 Decision Tree classifier was employed on training data. The result of this study performs satisfactory with consistent accuracy of 98.6%. Authors believe using this approach, road quality of selected zones can be visualized, thus can be possible to offer constructive feedback to drivers as well as to local authorities.

Inertial sensors such as accelerometer and gyroscope has been applied for a long period of time in aerospace applications but relatively less in robotics applications. In aerospace industry applied inertial system are comparatively too expensive for robotics applications. However, low-cost inertial systems are being developed for robotics systems which is motivated by automotive industry. Barshan et al. [89] proposed a low-cost solid-state gyroscope for robotics applications. For detecting the changes in position of a moving robot, an error model for the sensor including a Kalman filter was developed. The performance of orientation estimation in both cases – with error model and excluding error model – were verified. Result illustrates that the error measurement in localization without error compensation is between 5-15°/min. By applying an adequate error model, the performance, however, can be improved at least by a factor of 5. The findings of this approach suggests that with appropriate modeling of error sources, inertial sensors can be very useful in terms of delivering valuable information for mobile robotics applications.

3. DATASET DESCRIPTION

eSense platform, developed by Nokia Bell Labs, Cambridge, is the key component that has been used in this thesis to collect data.

Data collection was not part of this thesis work, it was done by others. Data set is taken from its owner with permission in order to do further analysis in this thesis. Therefore, in this dataset description section, the main focus is to provide an overview of eSense device, followed by description of data collection process. In addition, this section illustrates the process of data pre-processing as well as explain the method of features extraction. Apart from taking dataset from others, rest of the work such as data pre-processing, feature extraction are accomplished as a part of this thesis work.

The eSense device was used according to guidelines given in eSense User Documentation for data collection [90]. Therefore, in the following section information relates to procedure of using of eSense device is discussed.

3.1. ESense Platform

3.1.1. Platform Overview

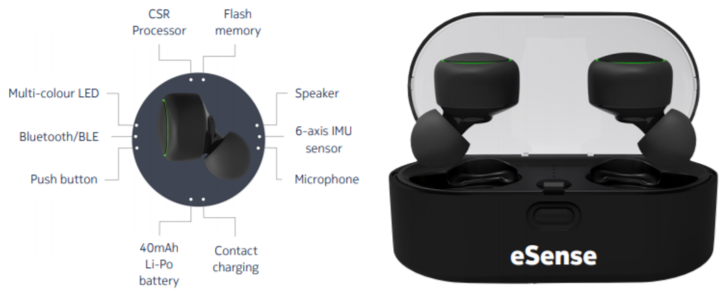


Figure 4. eSense Device [90]

This eSense device Figure 4 is a True Wireless Stereo (TWS) earbud equipped with 6-axis inertial motion unit, a microphone, as well as dual mode Bluetooth which are Bluetooth Classic and Bluetooth Low Energy. It is a multisensory earable platform that is employed for many research purposes specifically for personal-scale behavioral analytics.

The device – eSense – is developed with a customized 15 x 15 x 3 mm PCB and comprised of a Qualcomm CSR8670. It is also equipped with a dual-mode Bluetooth audio system-on-chip (SoC) including a microphone in each earbud. Additionally, it has a built-in InvenSense MPU6500 six-axis inertial measurement unit (IMU) comprising with a three-axis accelerometer, a three-axis gyroscope, and a two-state button. A circular LED, associated power regulation, and battery-charging circuitry are fitted as well in this device. It does not have any internal storage or real-time clock functionality. An ultra-thin 40-mAh LiPo battery is used to generate power for this device. The eSense earbuds can be recharged on the go using carrier case which has built-in battery facility. Each earbud is 20 g heavy, and the dimension is 18 x 20 x 20 mm.

The left earbud contains the IMU sensor which is accessible through BLE interface. Both earbuds have microphone functionality, and it is used by pairing Bluetooth Classic interface of the earbud with a host device. In this case of data collection, Samsung S10 5G mobile phone was the host device.

3.1.2. Pairing

In order to turn on the device, the push button located under the eSense logo needs to be pressed until LED turns to blue. If a single earbud is on and is not paired with any host device, the LED blink red, otherwise in normal situation LED blink blue. Both earbuds can be switched off by pressing and holding the push button of one earbud until the LED turns red.

Figure 5. Pairing [90]

Earbuds are kept in the carrier case and both earbuds are paired to each other in that case, thus only one earbud is required to connect with host device. Once the pairing procedure - connecting Bluetooth Classing interface of the earbud to a host device - is done, streaming music to earbuds and recording sound using microphones are achieved. Push button has to be pressed and hold until LED starts blinking red and blue to ensure the device is in pairing mode. If the pairing mode is done successfully, a device name with *eSense<Number>* Figure 5 pops up in the host device. By clicking the new device name, earbuds are paired, and earbuds start blinking blue to confirm the pairing. For best user experience, it is suggested to pair right earbud to the host device when both earbuds are employed. However, if the single earbud is in use, it is recommended to use the left one as it contains the IMU sensor and the BLE interface.

3.1.3. Wearing

As displayed in the image below Figure 6, firstly the earbuds needs to be inserted into the ear canal softly and then placement can be adjusted by rotating it slightly in order to wear the earbuds properly as well as to prevent them from falling off during head movements.

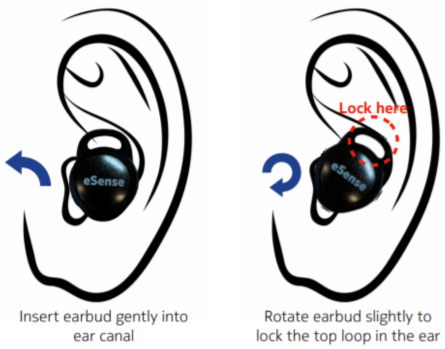


Figure 6. Wearing [90]

3.1.4. Motion and Proximity - BLE Interface

A BLE interface is equipped with left earbud, and it is used to configure the IMU sensor aiming to collect accelerometer and gyroscope data. It can also be used to detect proximity as left earbud transmit periodic BLE advertisement packets. Each left earbud is detected with a unique device name like *eSense-<four digits number>*. A BLE scanner application on a host device can easily identify the earbud's name. Nordic nRF connect is best possible option for Android and iOS, or BLE Scanner for iOS. The Figure 7 shows how setting up a scanning filter with the name of eSense can help to discover three earbuds as well as can assist to represent valuable information such as MAC address, RSSI and advertisement interval [90].

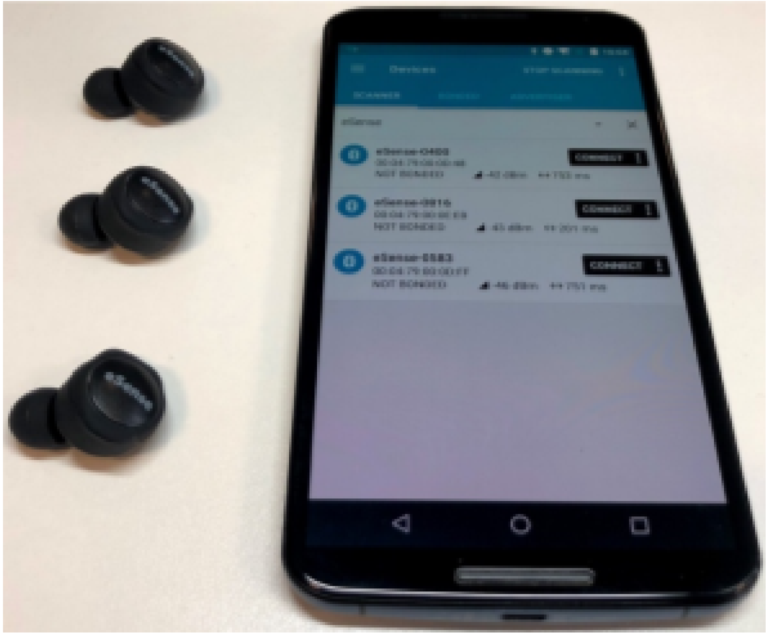


Figure 7. Motion and Proximity Sensing [90]

3.1.5. Orientation of IMU Axes

The orientation of the IMU axes as well as the polarity of rotation of the earbud is displayed below Figure 8.

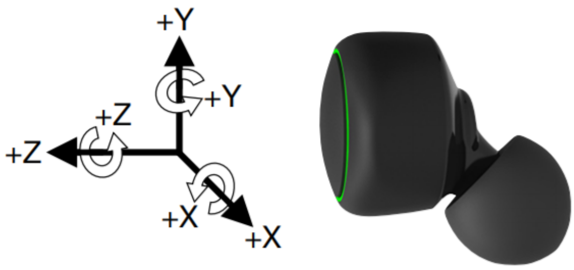


Figure 8. IMU axes [90]

Generally, IMU sampling starts with a certain sampling rate. Later, the accelerometer and gyroscope full scale range can be configured which defines the minimum and maximum acceleration and rotational speed detected by the sensor. Enabling onboard low pass filter for accelerometer and gyroscope is also possible [91].

IMU sensors particularly gyroscope produce a bias in their readings. Calibration eSense is allowed by this application and compensate for the bias. The earbuds required to be completely still and placed in a known position to identify the bias. To gain this place, the earbud needs to be upside down in the charging case as shown in the Figure 9.

Figure 9. Upside down positioning of eSense device [91]

Then calibration can be started by pressing the button in the UI. Once the calibration is done, messages pop up in the console displaying the bias value for each axis. When next time IMU sampling is activated, these values are automatically used to the IMU data.

When the earbud is in a known position, generally 200 accelerometer and gyroscope samples are collected and subsequently take the average of the values for each axis. This provides the offset that is required to be employed to each axis aiming to have the appropriate reading. For gyroscope X,Y, and Z very close to 0 when the earbud is still, for accelerometer X=0, Y=0, and Z=-1 when the earbud is facing down.

3.2. Data Collection Process

The dataset is originally given, collected via eSense device by others in a lab environment. It was not part of this thesis work. Primarily, user was given eSense devices and was instructed to wear it and move their head in sitting position. Subsequently, an app was used on an Android device to collect device reading. The app was tested on Android 9 particularly on Samsung S10 5G mobile phone. Five persons participated to the data collection process. Each participant performed the task six times moving his/her head continuously in different direction such as left, right, and straight. Each file of the dataset has about 1000 readings for different direction for a specific time period. All together there are 30 files in dataset.

The following Figure 10 displays the procedure of data collection using eSense device. To develop this figure, images are taken from Google which are publicly available to use for any purpose and a online free diagram editor [4] is used.

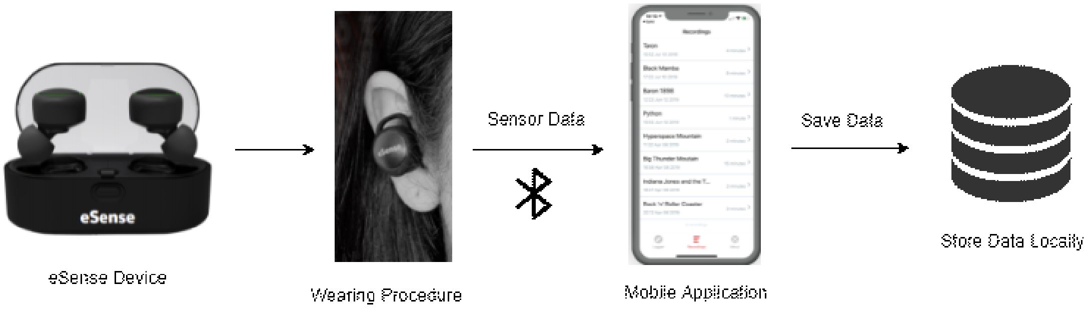


Figure 10. Procedure of data collection using eSense device

Readings collected via eSense device in datasets are provided below:

- Id (anonymous and unique user id)
- Timestamp (Time period of collected data)
- Ax (X-axis accelerometer values)
- Ay (Y-axis accelerometer values)
- Az (Z-axis accelerometer values)
- Gx (X-axis gyroscope values)
- Gy (Y-axis gyroscope values)
- Gz (Z-axis gyroscope values)
- Latitude
- Longitude
- Direction (Labelled - Straight, Left, Right)

[4]https://www.diagrameditor.com/

The Figure 11 shows a snap of dataset readings:

	id	TimeStamp	Ax	Ay	Az	Gx	Gy	Gz	Latitude	Longitude	Direction
0	0	0.000000e+00	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	NaN	NaN	NaN
1	204	1.596710e+12	-0.530518	0.815308	-0.259644	-128.018293	149.603658	-74.115854	65.057851	25.469975	L
2	205	1.596710e+12	-0.572632	0.773682	-0.258667	-126.829268	149.115854	-72.987805	65.057851	25.469975	L
3	206	1.596710e+12	-0.577026	0.768677	-0.259033	-115.548780	139.542683	-70.579268	65.057851	25.469975	L
4	207	1.596710e+12	-0.549072	0.796143	-0.257690	-113.262195	138.689024	-61.554878	65.057851	25.469975	L
...	...	...	...	...	...	...	...	...	...	...	...
974	220	1.596710e+12	-0.431519	0.882568	-0.356323	2.286585	2.835366	0.701220	65.057851	25.469969	S
975	221	1.596710e+12	-0.432617	0.884399	-0.356201	2.225610	2.743902	0.579268	65.057851	25.469969	S
976	222	1.596710e+12	-0.433472	0.886353	-0.356201	1.890244	2.073171	0.457317	65.057851	25.469969	S
977	223	1.596710e+12	-0.432495	0.886475	-0.355957	1.707317	1.707317	0.579268	65.057851	25.469969	S
978	224	1.596710e+12	-0.431519	0.885864	-0.356445	1.524390	1.341463	0.762195	65.057851	25.469969	S

Figure 11. Snap of Dataset Readings

3.3. Data Preprocessing

In data pre-processing step, very little job was required to be done. All the values in dataset readings were needed to achieve the objective of this study. Only values of latitude and longitude were eliminated as there are no changes in these values and have no significant purpose for further analysis.

3.4. Features Extraction

In order to extract features, each file of dataset is separated into three different files based on labeled direction. For example, each file has readings for accelerometer and gyroscope data for straight, left and right direction. Subsequently, based on directions, the file is separated into three files. The same process has been performed for all 30 files which means 90 separated csv files are created. The whole process is completed manually.

The steps are applied in this thesis for feature extractions:

- Main file of dataset having readings for all three directions (straight, left and right)
- Divided each csv file into three different csv files based on direction
- Used Jupyter Notebook (python libraries) to compute features
- Get the feature

In Python based Jupyter Notebook, libraries like Pandas, NumPy, Matplotlib were used for feature extraction. Considering the goal of this thesis, the following features has been extracted under statistical and time domain.

Table 3. Extracted Features

Domain	**Features**
Statistical	Mean, STD, Variance, Median, Min, Max, 25%, 50%, 75%, Range, Interquartile range (IQR), Kurtosis, Skewness, RMS (Root Mean Square)
Time	Integral, Auto-Correlation

These features were chosen for this study based on an similar analysis of accelerometer data conducted by Hemminki at al. [92]. In this paper, authors evaluated the effectiveness of these features and were able to detect different transportation mode on smartphones accurately.

The meaning of each feature is described below:

- Mean: Mean mainly indicates the average value, meaning it is the sum of all numbers divided by the number of numbers. Similarly, the mean of a sample $\{x_1, x_2, \ldots, x_n\}$ usually signified by $\overline{x}$, which means it is the sum of the sampled values divided by the number of items in the sample [93].

  $$\bar{x} = \frac{1}{n}\left(\sum_{i=1}^{n} x_i\right) = \frac{x_1 + x_2 + \cdots + x_n}{n}$$

  For example, if the values are: 4, 36, 45, 50, 75. Then mean is 42.

  $$\frac{4 + 36 + 45 + 50 + 75}{5} = \frac{210}{5} = 42.$$

- Variance: Variance refers to the average of the squared differences from the mean. In another term, variance is used to measure how far a set of numbers is spread out from their average value. Variance has an important role in the field of science, where statistical analysis of data is common.

  Generally, two types of formulas are used to measure the variance based on the method of collected data. Population variance is applied if the data is collected from every member of the population, whereas sample variance is employed if the data is gathered from samples. For this thesis, sample variance is applied to calculate variance of specific columns using Pandas DataFrame [94].

The formula is:

$$s^2 = \frac{\Sigma (X - \bar{x})^2}{n - 1}$$

Where S^2 = Sample variance
x_i = the value of the one observation
$\overline{x}$ = the mean value of all observations
n = the number of observations

- STD: The standard deviation is a calculation of amount of variation of a given set values. A low standard deviation means the values seems to be close to the mean, whereas a high standard deviation depicts the values are spread out over a wider range. In Pandas library, standard deviation is abbreviated by STD [95].

  Two formulas for Standard Deviation are given below [96]:

  The population standard deviation:

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}$$

  The sample standard deviation:

$$s = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \overline{x})^2}$$

  The major difference is divided by $N - 1$ instead of N if sample variance is calculated.

- 25%: The 25th percentile refers to the value at which 25% of the answers lie below that value.

- 50%: It also known as median. The median cuts the dataset into half where half of the answers lie below median, and half are above median.

- 75%: It also known as third quartile where 75% answers sit below that value.

  The 25th percentile is also called as the first quartile $(Q1)$, the 50th percentile as the median or second quartile $(Q2)$, and the 75th percentile as the third quartile$(Q3)$ [97].

In order to achieve different percentile, the data is split into quarters [98]. For example, if the data numbers are 1, 3, 3, 4, 5, 6, 6, 7, 8, 8, then

$$Q1 = 3$$
$$Q2 = 5.5$$
$$Q3 = 7$$

In this thesis, different percentile values are achieved by Pandas *df.describe()* command.

- Min: The min (minimum) simply represents the lowest observation.
- Max: Max (maximum) is the largest observation of sample, means the values of the greatest elements of sample.

  Calculating the min and max helps to understand the total span of the data. Additionally, it helps to get more familiar with data. The reason to calculate the min and max might vary depending on the purpose of the study [99].

- Range: The range value refers to the difference between lowest and highest values. It is a numerical indication of the span of the data. Range can be calculated by subtracting the min from the max [99].

$$Range = Max - Min$$

- Median: Median represents the middle number. Median is the value that separates the higher half from the lower half of a dataset.

  In other words, when numbers are finite and listed in order from smallest to greatest, the median is the middle number [100].

  If the dataset has an odd number of observations, the middle is selected as the median. For example, if the numbers are 1, 3, 3, 6, 7, 8, 9 then the median is 6.

  In general, for a set x of n, the formula is:

$$median(x) = x_{(n+1)/2}$$

  In case of dataset has no distinct middle value, then the median is calculated by measuring the mean value of two middle values. Such as if the dataset is 1, 2, 3, 4, 5, 6, 7, 8, 9 then the median is: 4.5 that is (4+5)/2.

- Skewness: Skewness is mainly the measurement of symmetry, or to be more specific, the lack of symmetry. A dataset is symmetric if it seems the same to the left and right of the center point [101].
- Kurtosis: Kurtosis is all about the tails of the distribution which measures the tail-heaviness of the distribution. Kurtosis is useful to outline whether the data is heavy-tailed or light-tailed relative to a normal distribution [101].

Histogram is an effective graphical method to display both the skewness and kurtosis of a data set.

For sample of n values, skewness formula is [102]:

$$g_1 = \frac{m_3}{m_2^{3/2}} = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^3}{\left[\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^2\right]^{3/2}}$$

where x is the sample mean, s is the sample standard deviation, m_2 is the (biased) sample second central moment, and m_3 is the sample third central moment.

For sample of n values, kurtosis formula is [103]:

$$g_2 = \frac{m_4}{m_2^2} - 3 = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^4}{\left[\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^2\right]^2} - 3$$

where m_4 is the fourth sample moment about the mean, m_2 is the second sample moment about the mean, x_i is the ith value which is sample value, and x is the sample mean.

The Figure 12 below illustrates the skewness and kurtosis in a graphical view:

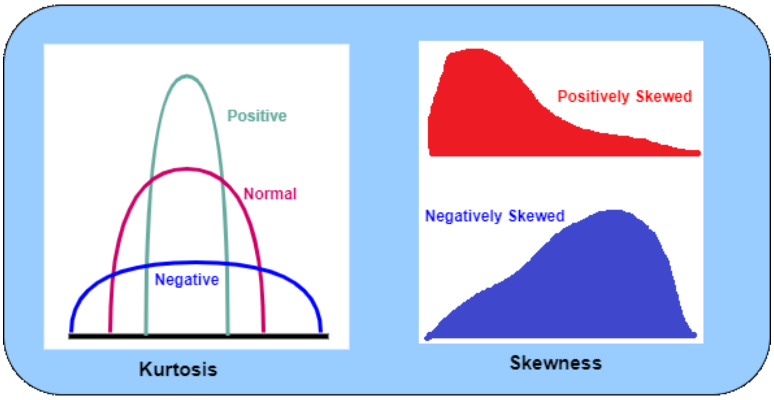


Figure 12. Kurtosis and Skewness

- Interquartile Range (IQR): The IQR is the measurement of where the bulk of the values sit. It mainly describes the spread of the middle half of the distribution.

  Quartiles does the segmentation job for any distribution that is ordered from low to high into four equal parts. The interquartile represents the second and third quartiles, or the middle half of the data set [104].

Formula for interquartile range is:

$$IQR = Q3 - Q1$$

- Root Mean Square (RMS): It is defined as the square root of the mean square. Applying following steps, the root mean square can be calculated as below [105]:
  - Firstly square all the values
  - Then take the average of the squares
  - Lastly take the square root of the average

  In the case of a set on n values $x_1, x_2, \ldots, x_n$, the RMS is [106]:

  $$x_{\mathrm{RMS}} = \sqrt{\frac{1}{n}\left(x_1^2 + x_2^2 + \cdots + x_n^2\right)}.$$

  Where x_i is the items under observations and n is the total number of items.

- Integral: Integral assign numbers to functions that helps to describe displacement, area, volume, and other ideas occurred by combining infinitesimal data. The process that is used to find integrals is call integration. Integration, along with differentiation, is considered as a fundamental, essential operation of calculus. Integration serves as a tool to solve various problem in different field of science such as physics, mathematics, statistics, and so on [107].

  In this thesis, integrate library from SciPy - a Python based ecosystem open-source software - was imported to perform the integration on specific columns of Pandas dataframe.

- Autocorrelation: Autocorrelation is the similarity between observations as a function of the time lag between them. It is commonly performed in time domain signals to analyze functions or series of values, such as time domain signals [108].

  Considering the measurements, $y_1, y_2, \ldots, y_n$ at time $x_1, x_2, \ldots, x_n$, the lag k autocorrelation function is defined as

  $$s_k = \frac{1}{n}\sum_{i=1}^{n-k}(y_i - \bar{y})(y_{i+k} - \bar{y}) = \frac{1}{n}\sum_{i=k+1}^{n}(y_i - \bar{y})(y_{i-k} - \bar{y})$$

For feature extraction, only gyroscope sensor data was considered, the accelerometer sensor data was avoided because there is very minimal changes in acceleromete data. To illustrate, accelerometer and gyroscope data for whole dataset is visualized in the following Figure 13, Figure 14, respectively:

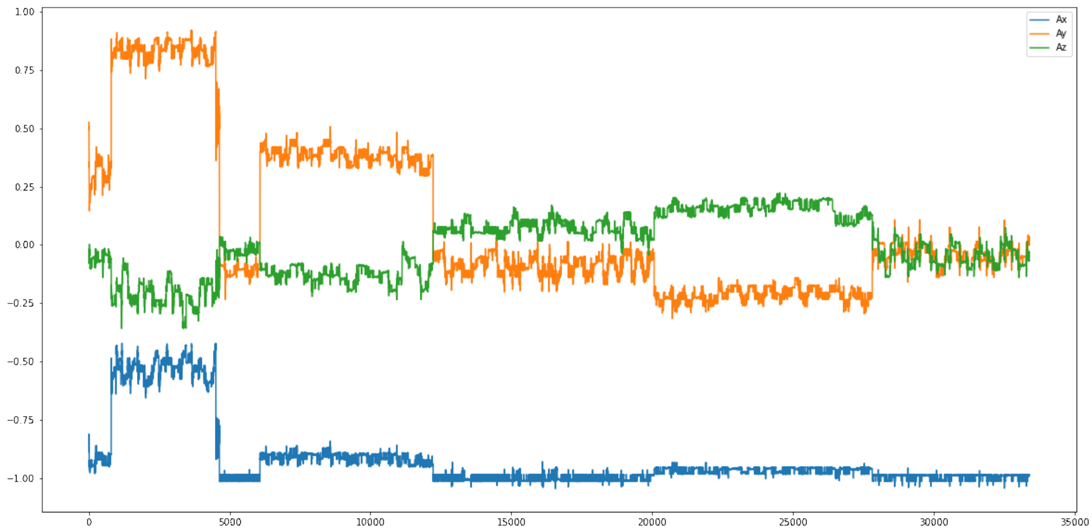


Figure 13. Accelerometer Data

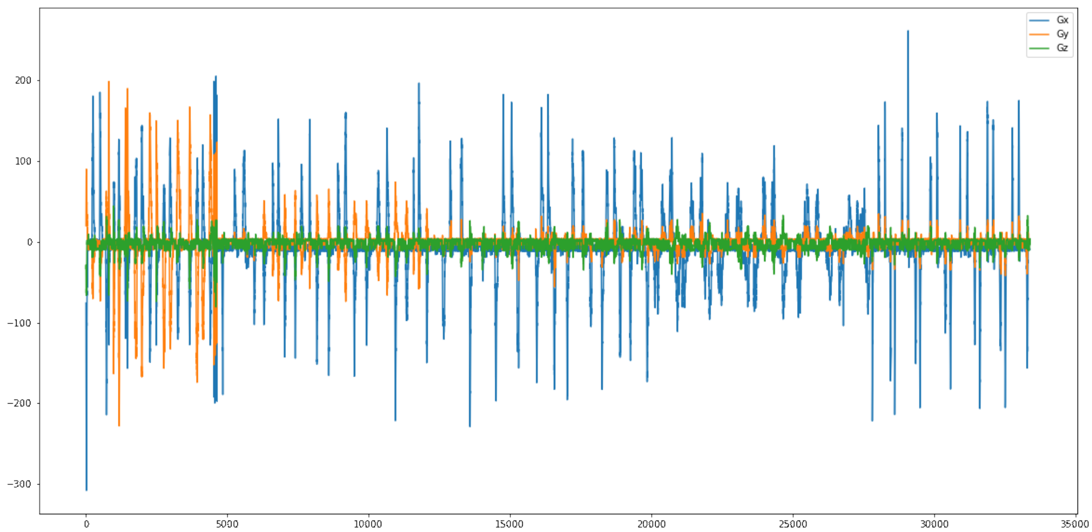


Figure 14. Gyroscope Data

Once the features extraction for each file was done, which means for 90 csv files altogether, using Jupyter Notebook, subsequently all files were integrated into one single csv file. To do this, Pandas *pd.concat(frame)* command was applied using Jupyter Notebook.

The data of final csv file after completing the feature extraction is displayed through a line graph in Figure 15 :

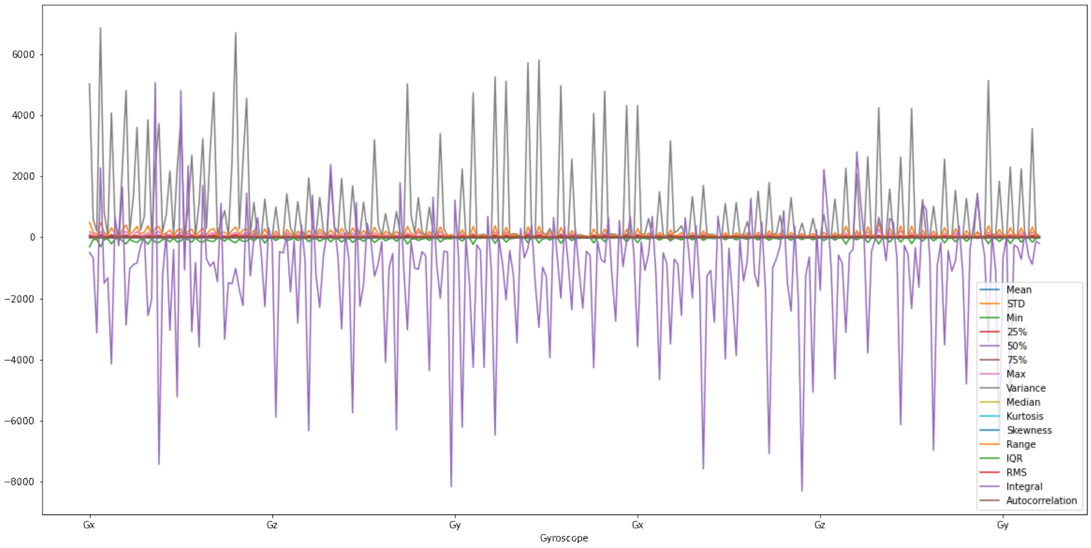


Figure 15. Line graph of feature extraction data

4. THEORETICAL BACKGROUND

In this section the theoretical background of machine learning models that have been applied for the purpose of this thesis is described. The goal of this thesis is to detect head direction - left, right, or straight - applying these machine learning models. The four machine learning models are Random Forest, Support Vector Machine, Naïve Bayes, and Perceptron. To evaluate the performance of all these models, the accuracy, precision, recall, and f1 score measurement metrics have been applied.

Machine Learning (ML) is a computational algorithm that is designed to develop methods aiming to apply to automatically recognize patterns in data and using those patterns future data or other desired outcome can be predicted [109]. The new era of technology is driven by big data and machine learning models are considered as working horse of this new era of big data. Machine learning techniques have been applied in a wide range of fields including pattern recognition, computer vision, spacecraft engineering, finance, entertainment, and computational biology to biomedical and healthcare applications [110]. Among all these applications, the two major fields of machine learning applications are predictive data analysis and data mining. ML develops models mainly using training data to make predictive assessments.

4.1. Random Forest

Random forests invented by L. Breiman in the early 2000s [111] is one of the most successful methods currently available to handle sheer volume of modern data sets. This supervised learning procedure practices simple but effective principles - divide and conquer - to operate. Firstly, it creates sample fraction of the data, then grow randomized tree predictor on each small piece, subsequently aggregate these predictors together [112].

To illustrate, the random forest algorithm follows four key steps to perform.

- First, it selects random samples from a provided dataset.
- Second, it builds a decision tree for each sample and obtain a prediction result from each of the decision tree.
- Then a vote is performed for each predicted result.
- Lastly, for final prediction, it takes the prediction result with most votes.

The diagram below explains all the steps:

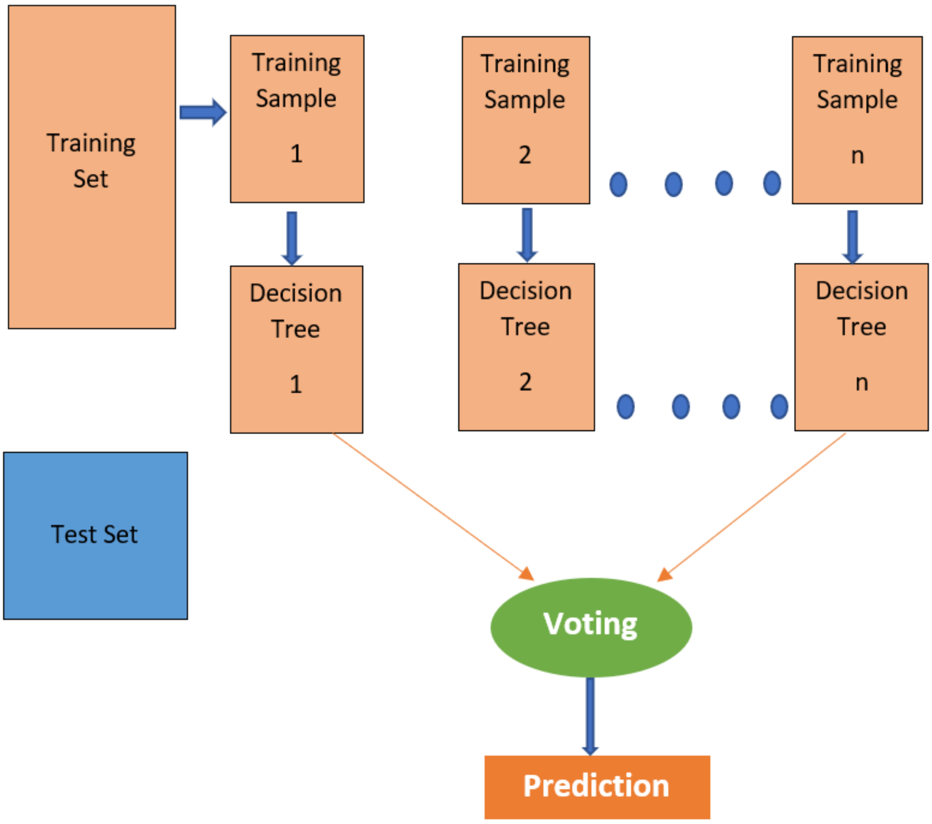


Figure 16. Random Forest

According to the paper of Fawagreh et al. [113], the tree of random forest following the rules are given below:

- If the number of cases in training dataset is N, then N is used as training data for growing the tress as randomly selected sample.
- If the number of variables is M, the variable m is selected considering the condition of m«M for each node.
- From M, m is randomly chosen.
- Best method to split is utilized to m.
- The tree is grown as much as possible.

For the purpose of this thesis, the Scikit-Learn python libraries for random forest ensemble classifier [5] has been applied. One of the key benefits of using RF is to find out best features. For this thesis, understanding the most important features for detecting head direction is essential. The RF model estimates which features are being considered repeatedly in different decision trees for predicting final results in order find best feature.

[5]1. Random Forest: https://scikit learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

4.2. Support Vector Machine

Support Vector Machine (SVM) is a supervised learning method largely used for classification and regression. In other words, support vector machine is a classification and regression prediction tool that applies machine learning theory aiming to achieve highest predictive accuracy, while it is capable of avoiding overfitting problem to the data automatically [114]. SVM is able to manage multiple continuous and categorical variables. To separate different classes, SVM builds a hyperplane in multidimensional space. Optimal hyperplane in an iterative manner is produced by SVM that is utilized to lessen error. The fundamental concept of support vector machine is to identify a maximum marginal hyperplane (MMH) to divide the data sets into classes.

The SVM algorithm uses a kernel for implementation. It is mostly known as kernel trick for SVM. A kernel is employed to transform an input data space into essential form. Kernel deals with transforming a low-dimensional input space to a high dimensional space. In another terms, it does the job of converting non-separable problem to separable problems, adds more dimension to it to do this converting job. It is most beneficial to deal with non-linear separation problem. Kernel tricks is useful to develop an accurate classifier.

This thesis applies Scikit-learn implementation [6] using python libraries. Scikit-learn offers three types of kernel options for different purposes.

- Linear Kernel: This kernel can be employed as normal dot product any two provided observations. The sum of the multiplication of each pair of input values is the product between two vectors.
- Polynomial Kernel: It is considered as a more generalized form of linear kernel which is capable of distinguishing curved and nonlinear input space.
- Radial Basis Function Kernel: This RBF kernel is very famous kernel and is commonly applied in SVM classification. It is useful in mapping an input space to infinite dimensional space.

The Scikit-learn implementation is used for this thesis which formulates mathematical calculation in following way:

Considering training vectors $x_i \in R^p$, $i = 1, \ldots, n$ in two classes, and a vector $y \in \{1, -1\}^n$, the aim is to identify $w \in R^p$ and $b \in R$ to ensure prediction is correct for most sample given by sign $(w^T \phi(x) + b)$.

[6] Support Vector Machine: https://scikit-learn.org/stable/modules/svm.html

Support Vector Machine is capable of solving the following primal problem:

$$\min_{w,b,\zeta} \frac{1}{2} w^T w + C \sum_{i=1}^{n} \zeta_i$$
$$\text{subject to} \quad y_i(w^T \phi(x_i) + b) \geq 1 - \zeta_i,$$
$$\zeta_i \geq 0, i = 1, \ldots, n$$

Intuitively, the aim is to maximize the margin by minimizing $||w||^2 = w^T w$, while incurring a penalty in a situation like a sample is misclassified or within the margin limit. A perfect prediction is indicated if the value $y_i(w^T(x_i) + b) \geq 1$ for all samples. However, with a hyperplane, the problems are not always separable, thus some samples are allowed to be at distance ζ_i from their correct margin limit. The strength of this penalty is controlled by the penalty term C, therefore it acts as an inverse regularization parameter, explained below:

The dual problem to the primal is

$$\min_{\alpha} \frac{1}{2} \alpha^T Q \alpha - e^T \alpha$$
$$\text{subject to} \quad y^T \alpha = 0$$
$$0 \leq \alpha_i \leq C, i = 1, \ldots, n$$

where e signifies as vector of all ones, Q signifies as n by n positive semidefinite, $Q_{ij} \equiv y_i y_j K(x_i, x_j)$, and the kernel is $K(x_i, x_j) = \phi(x_i)^T \phi(x_j)$. The terms α_i are called the dual coefficients, and they are upper-bounded by C. This dual representation highlights the fact that training vectors are implicitly mapped into a higher (maybe infinite) dimensional space by the function ϕ.

After solving the optimization problem, for a given sample x, the output of a decision function becomes like this:

$$\sum_{i \in SV} y_i \alpha_i K(x_i, x) + b,$$

and the predicted class relate to its sign. It is required to sum over the support vectors as the dual coefficients α_i are zero for the other samples.

These parameters are accessible through the attributes *dual_ coef_* that is holding the product $y_i \alpha_i$, *support_ vectors_* which holds the support vectors, and *intercept_* that possess the independent term b.

4.3. Naive Bayes

Naïve Bayes methods are a set of supervised learning algorithms which use Bayes' theorem and "naïve" assumption. Naïve assumption is an assumption of conditional independence between every pair of features given the value of the class variable.

The Scikit-learn implementation is used for this thesis which formulates mathematical calculation in following way [7]:

Bayes' theorem states the relationship given below, considering the given class variable is y and dependent feature vector is x_1 through x_n:

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y)P(x_1, \ldots, x_n \mid y)}{P(x_1, \ldots, x_n)}$$

If the conditional independence assumption is used, then

$$P(x_i|y, x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) = P(x_i|y),$$

for all i, this relationship can be defined to

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y)\prod_{i=1}^{n} P(x_i \mid y)}{P(x_1, \ldots, x_n)}$$

Since $P(x_1, \ldots, x_n)$ is constant given input, following classification rule can be applied:

$$P(y \mid x_1, \ldots, x_n) \propto P(y)\prod_{i=1}^{n} P(x_i \mid y)$$
$$\Downarrow$$
$$\hat{y} = \arg\max_y P(y)\prod_{i=1}^{n} P(x_i \mid y),$$

Maximum A Posteriori (MAP) estimation can also be used to calculate $P(y)$ and $P(x_i|y)$; the former is then the relative frequency of class y in the training set.

There are different types of Naïve Bayes classifiers such as Gaussian, Multinominal, Complement, Bernoulli, Categorical, and Out-of-core naïve bayes model fitting. This differentiation in classifiers happens due to the assumptions they make regarding the distribution of $P(x_i|y)$.

[7]Naïve Bayes: https://scikit-learn.org/stable/modules/naive_bayes.html

Out of huge applications of Naïve Bayes classifiers in real-world situations, it is famously applied in the field of document classification and spam filtering. For Naïve Bayes, small amount of training data is required to estimate the necessary parameters. Thus, this classifier is comparatively very faster than other sophisticated methods. Using this classifier can help to alleviate the problem of dimensionality as well as it can estimate each distribution independently as a one-dimensional distribution.

For this study, Gaussian Naïve Bayes classifier is employed using scikit learn, and the algorithm for this specific classifier is given below:

$$P(x_i \mid y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right)$$

The parameters σ_y and μ_y are estimated using maximum likelihood.

4.4. Perceptron

Perceptron is considered as one of the first and simplest types of artificial neural networks. Despite the fact of not being a deep learning, it is an important building block. It is a linear learning algorithm for binary classification tasks.

Perceptron can quickly learn a linear separation in feature space for the purpose of two-class classification tasks. It applies stochastic gradient descent optimization algorithm to learn, and it does not predict calibrated probabilities.

The algorithm comprises of a single node or neuron that consider a row of data as input, subsequently a class label is predicted. This objective is achieved by estimating the weighted sum of the inputs and a bias that is set to 1. The weighted sum of the input of the model is named as the activation [8].

- **Activation** = Weights * Inputs + Bias

The output of the model is 1.0, if the activation is above 0.0. Otherwise, the model will output 0.0.

- **Predict1:** if activation > 0.0
- **Predict0:** if activation <= 0.0

Considering that inputs are multiplied by model coefficients, it is fair practice, like linear regression and logistic regression, to normalize and standardize data before applying the model. As scikit-learn implementation algorithm [9] using Python machine learning libraries is applied for this thesis, more information regarding mathematical formulation of this method can be found in scikit-learn documentation page.

[8] https://machinelearningmastery.com/perceptron-algorithm-for-classification-in-python/

[9] Perceptron: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Perceptron.html

4.5. Methods for Evaluation

Evaluation refers to the process that is used to analytically examines a program. It is a process comprised with collecting and analyzing information regarding a program's activities, characteristic, and results. The key goal of using evaluation is to make assessment about a program, find out the shortcomings to improve its effectiveness, as well as to inform programming decisions based on analyzed outcomes [115]. Evaluation helps to understand program's success or progress. There are different types of evaluation methodology, thus evaluation procedure is decided based on the intended purpose of it, subsequently the gathered information should be presented in beneficial and effective manner to the decision makers. For this thesis, the outcome evaluation method is applied to investigate to what extent program is gaining its results. To assess the performance of applied machine learning models aiming to predict head movement, accuracy, precision, recall, and f1-score performance metrics have been applied for all models.

- **Accuracy**

  Accuracy measurement is an alternative way to convey information to identify the capability of certain forecasting models to predict actual data, in a condition like either a model is fitted to such data, or for future periods whose values have been unused to build the forecasting model [116]. In short, it is one of the most intuitive performance measurement metrics, and it mainly calculates the percentage of correct predictions for the test data.

$$\textbf{Accuracy} = \frac{\text{correct prediction}}{\text{all predictions}}$$

- **Precision**

  Precision calculates the ratio of correctly predicted positive observations to the total predicted positive observations [117].

$$\textbf{Precision} = \frac{\text{true positives}}{\text{true positives + false positives}}$$

- **Recall**

Recall estimates the ratio of correctly predicted positive observations to all observations in actual class [117].

$$\textbf{Recall} = \frac{\text{true positives}}{\text{true positives + false negatives}}$$

- **F1 Score**

F1 score takes both false positives and false negatives into account as it weights average of Precision and Recall. F1 score is considered as a better measurement metric if a balance between Precision and Recall is required and there is uneven class distribution.

$$\textbf{F1 score} = 2 \text{ x } \frac{\text{precision * recall}}{\text{precision + recall}}$$

5. EXECUTION

In order to achieve the goal of detecting head movement using gyroscope data collected through earables, the execution procedure was done following few key steps. Jupyter Notebook [10] which support Python language and comprised with different Python libraries such as Pandas [11], NumPy [12], Matplotlib [13] have been applied for whole implementation part in this study.

The reason to choose Jupyter Notebook is its a web-based interactive environment that combines code, rich text, images, videos, animations, mathematical equations, plots, maps, interactive figures and widgets, and graphical user interfaces, into a single document. This tool is capable of doing high-performance numerical computing and data analysis in Python. The prime advantage of this tool is it can produce result immediately cell by cell once command is given. Python code in Jupyter Notebook is saved as IPython file [14].

Crucial steps that have been performed in implementation are extracting features under statistical and time domain, apply four machine learning models using extracted features to detect the direction of head – left, right or straight. Subsequently, evaluate the performance of applied ML models by four evaluation metrics such as accuracy, precision, recall, and f1 score. Finding the importance of each features has also been achieved in this thesis.

5.1. Program Pipeline

5.1.1. Extraction of Features

For features extraction, once data pre-processing is accomplished, each csv file (30 files) that contains readings about id, timestamp, gyroscope data, and labeled direction are segmented into three csv files manually (total 90 csv files) based on labeled direction. After that Pandas library is employed to convert each csv file into Pandas Data frame using Jupyter Notebook to perform feature extraction. The Figure 17 given below represents the pipeline of whole procedure of features extraction:

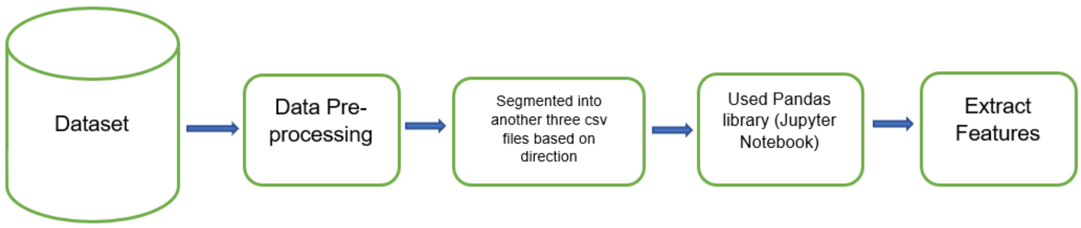
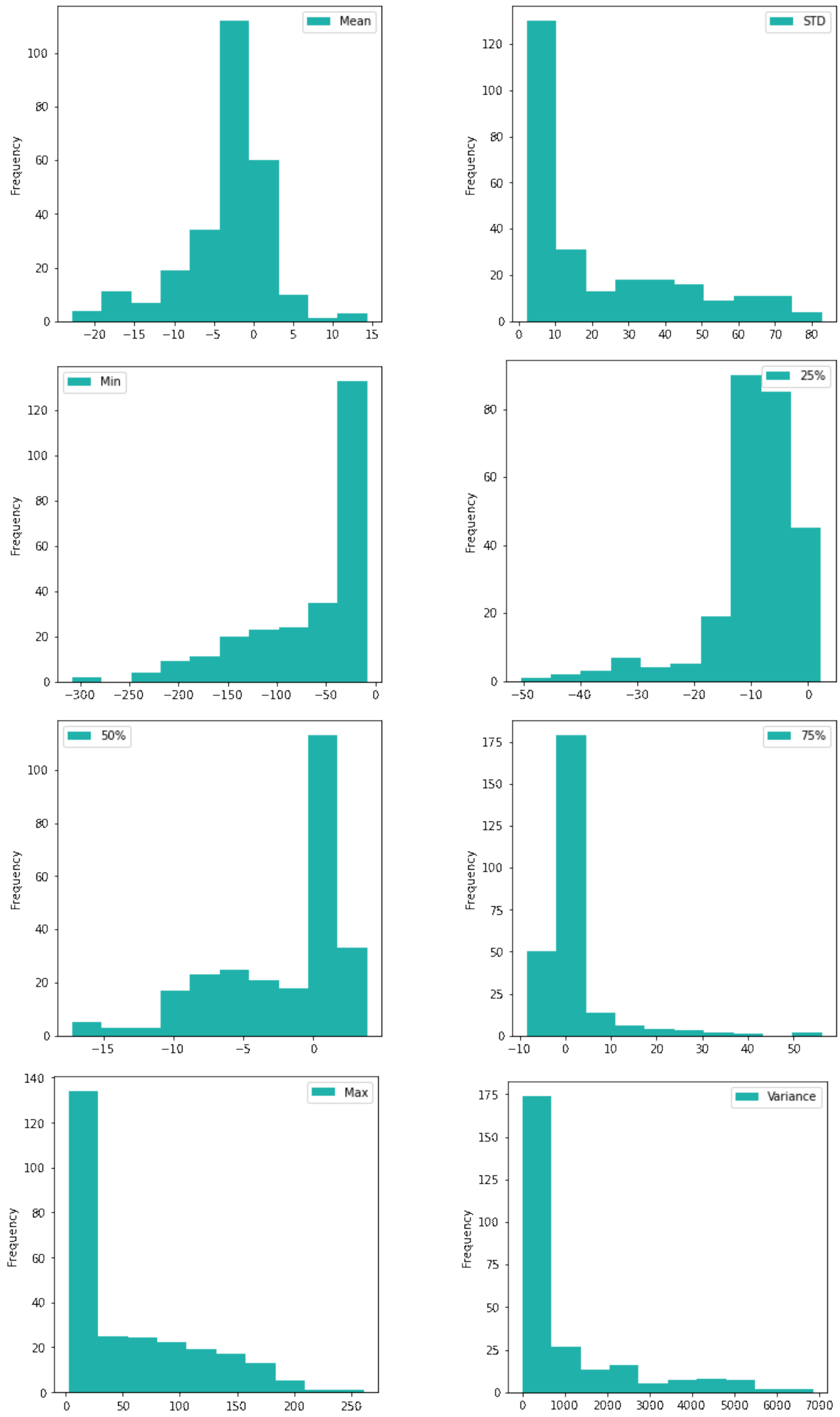


Figure 17. Feature extraction procedure pipeline

[10] Jupyter: https://jupyter.org/
[11] Pandas: https://pandas.pydata.org/docs/user_guide/index.html
[12] NumPy: https://numpy.org/doc/stable/user/index.html
[13] Matplotlib: https://matplotlib.org/stable/users/index.html
[14] IPython Cookbook: https://ipython-books.github.io/

In order to provide a concrete view to the feature data, distribution graph for each of the feature attribute is given in Figure 18.



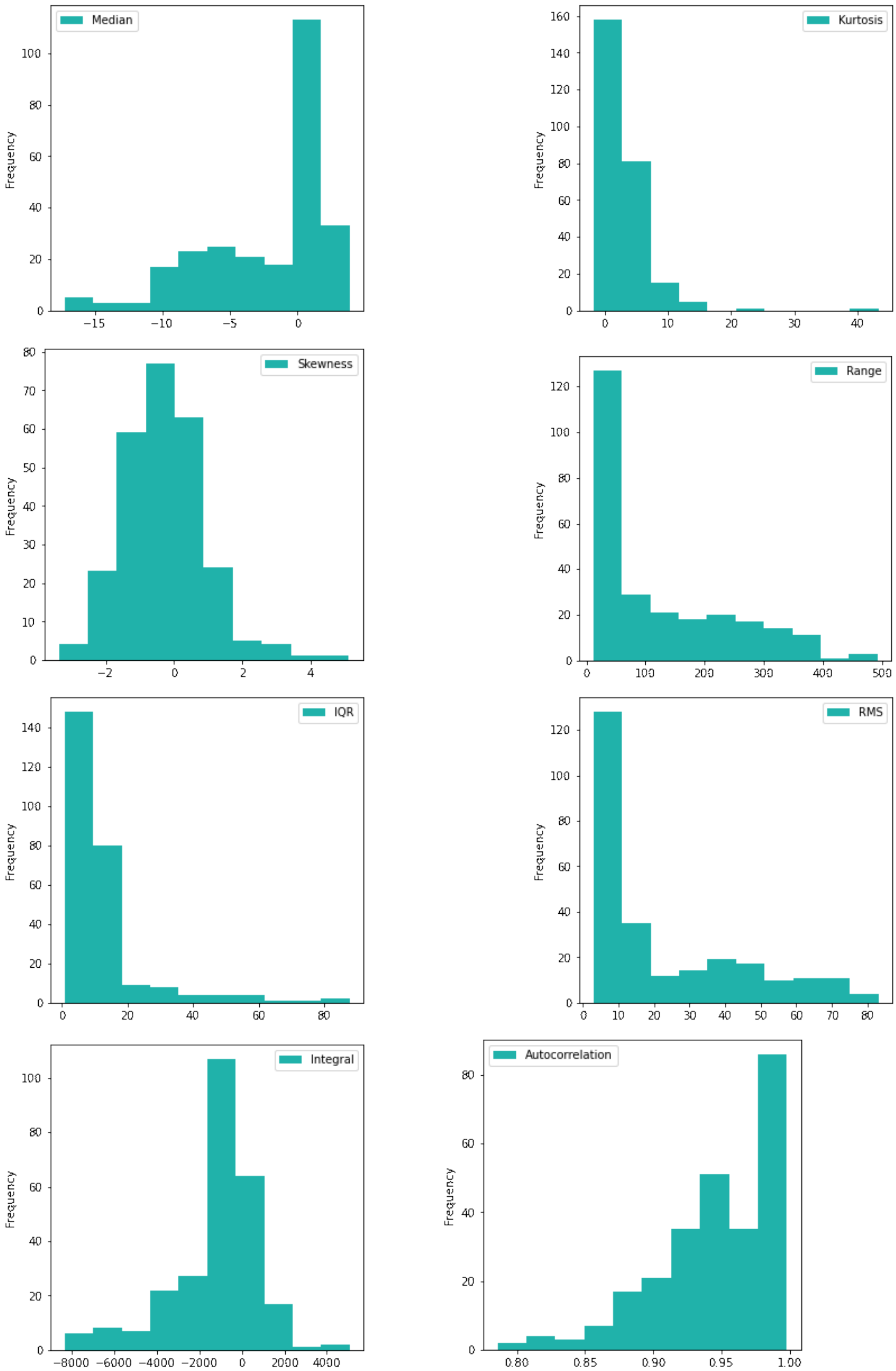


Figure 18. Distribution of each attribute of extracted features

Earlier in the section 3.4, adequate information about each feature attribute is provided, thus here in Table 4 the summarization of each feature attribute is delivered.

Table 4. Summarization of features attributes

Attribute Name	**Domain**	**Description**
Mean	Statistical	Indicates the average value
Variance	Statistical	The average of squared differences from the mean
Standard Deviation (STD)	Statistical	The amount of variation of a given set values
25%	Statistical	The first quartile
50%	Statistical	The second quartile
75%	Statistical	The third quartile
Min	Statistical	The lowest observation
Max	Statistical	The largest observation
Range	Statistical	The difference between lowest and highest values
Median	Statistical	The middle number
Skewness	Statistical	The measurement of symmetry
Kurtosis	Statistical	The measurement of tail-heaviness of the distribution
Interquartile Range (IQR)	Statistical	The measurement of where the bulk of the values sit
Root Mean Square (RMS)	Statistical	The square root of the mean square
Integral	Time	The description of displacement, area, volume, and other ideas
Autocorrelation	Time	The similarity between observations

5.1.2. Machine Learning Models Development

Since the feature extraction is achieved, these values are applied in machine learning models. In order to do so, scikit-learn libraries are employed to split the data into train and test category. For training 70% of data is used and 30% data is used for testing. Different classification models are imported from sklearn libraries. Earlier in this thesis, the official source of sci-kit learn libraries for machine learning models is provided in these sections 4.1, 4.2, 4.3, and 4.4.

To train the machine learning models, features; 'Mean', 'STD', '25%', '50%', '75%', 'Max', 'Variance', 'Median', 'Kurtosis', 'Skewness', 'Range', 'IQR', 'RMS', 'Integral', 'Autocorrelation' are applied as X value and for Y value the labeled direction - Straight, Left, Right - is utilized. Various evaluation metrics such as Accuracy, Precision, Recall, and F1-score have also been imported from scikit-learn libraries that is latterly applied to evaluate the performance of ML models. The description of each evaluation metric is delivered in this section 4.5.

The Figure 19 demonstrates the pipeline of machine learning models development and performance evaluation:

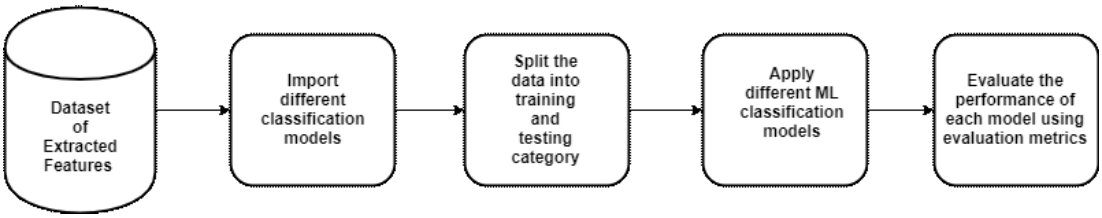


Figure 19. Pipeline of Machine Learning Model Development

5.1.3. Feature Importance

In predictive modeling, feature importance scores play a significant role. Identifying feature importance scores helps to acknowledge the insight of the data, insight of the model. It also helps to understand the basis for dimensionality reduction as well as feature selection. By doing so, the efficiency and effectiveness of a predictive model can be improved [118].

The random forest implementation of sklearn provides the functionality of *feature_importance_* which helps to determine the predictive power of each feature in given dataset. Therefore, taking into account the advantage that RF model offers, for this thesis, Random Forest *feature_importance_* property is applied to assess the importance of each feature. The outcome helps to recognize which feature contribute most to the decision of making predictive model.

6. RESULT

According to roadmap, the head movement detection is done by employing four different machine learning models – Random Forest, Support Vector machine, Naïve Bayes, and Perceptron. Consequently, the performance of these models is evaluated by four different evaluation metrics such as accuracy, precision, recall, and f1 score. Machine learning models are applied based on the extracted features.

The Table 5 demonstrates the accuracy score of four machine learning models:

Table 5. Accuracy score of four applied machine learning models

Model	**Accuracy**
Random Forest	0.7721518987341772
Support Vector Machine (SVM)	0.4177215189873418
Naïve Bayes	0.4050632911392405
Perceptron	0.3924050632911392

Accuracy score indicates that among all the models, Random Forest perform best with approximately 77% accuracy. All other models perform very close to each other, where SVM, Naïve Bayes, and Perceptron achieved about 42%, 40%, and 39% accuracy.

To visualize the accuracy result of different models, the following Figure 20 is given:

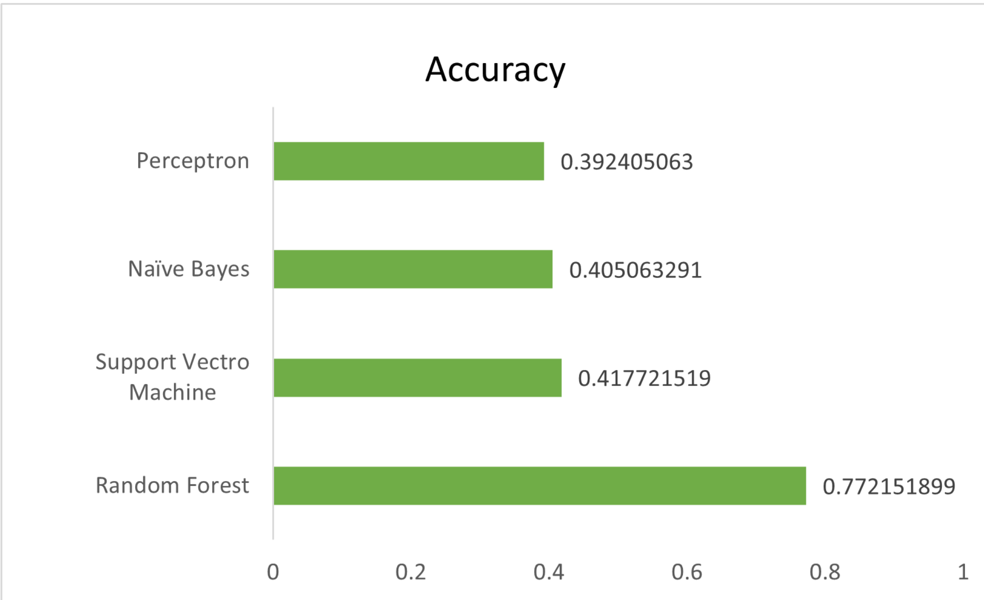


Figure 20. Accuracy score of different ML models

The Table 6 below provides the result for Precision, Recall, and F1 score of different models based on directions.

Table 6. Precision, Recall, and F1 score for applied machine learning models

Model	**Direction**	**Precision**	**Recall**	**F1 score**
Random Forest	Left	0.81	0.68	0.74
	Right	0.65	0.88	0.75
	Straight	0.92	0.76	0.83
Support Vector Machine (SVM)	Left	0.19	0.12	0.15
	Right	0.37	0.64	0.47
	Straight	0.70	0.48	0.57
Naive Bayes	Left	0.25	0.08	0.12
	Right	0.50	0.28	0.36
	Straight	0.40	0.79	0.53
Perceptron	Left	0.44	0.64	0.52
	Right	0.33	0.44	0.38
	Straight	0.40	0.14	0.21

In following Figure 21, Figure 22, Figure 23, Figure 24, the performance of Random Forest, Support Vector Machine, Naive Bayes, and Perceptron models are visualized through bar graph, respectively.

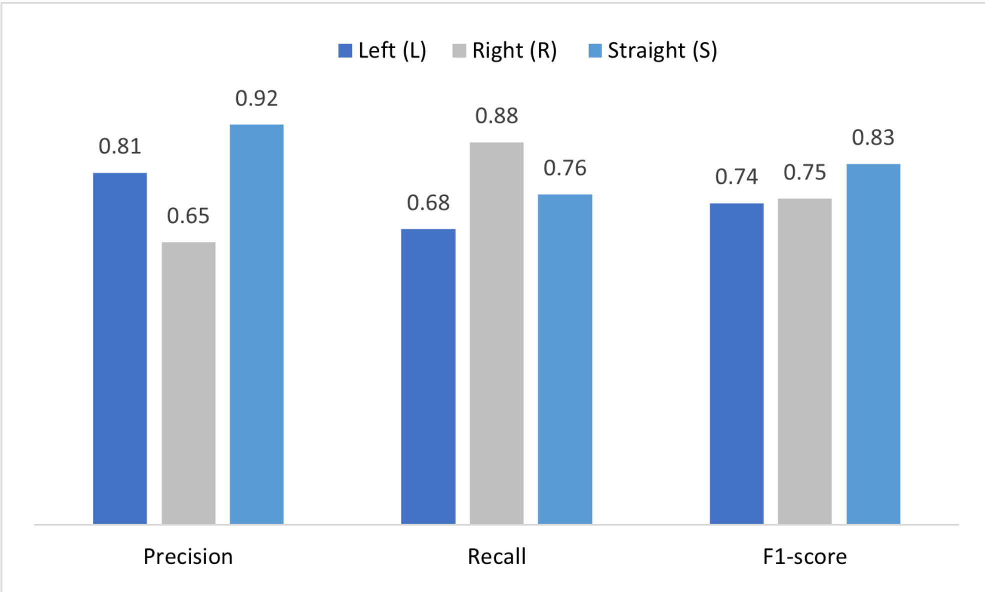


Figure 21. Performance of Random Forest

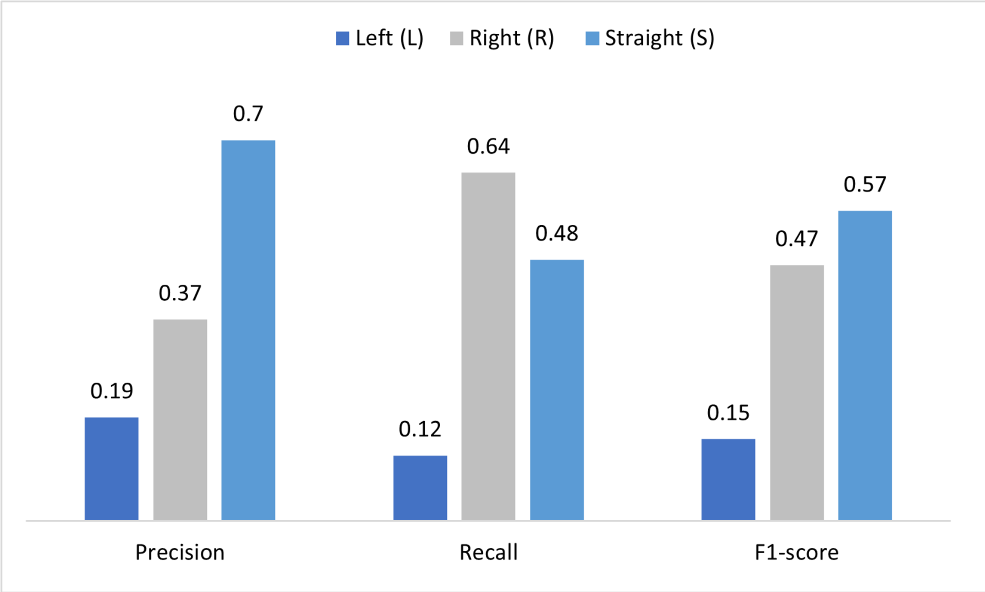


Figure 22. Performance of Support Vector Machine

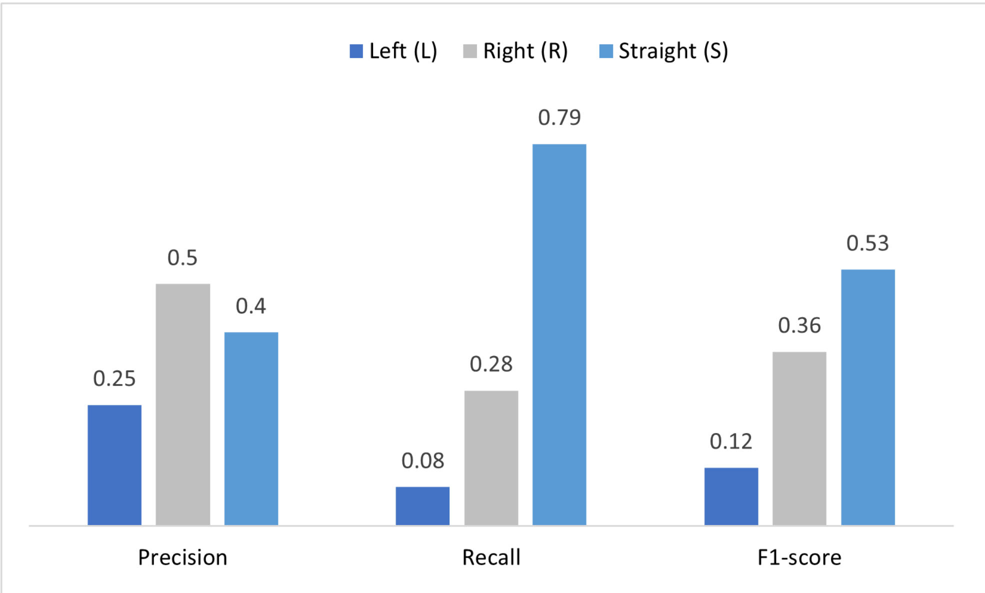


Figure 23. Performance of Naive Bayes

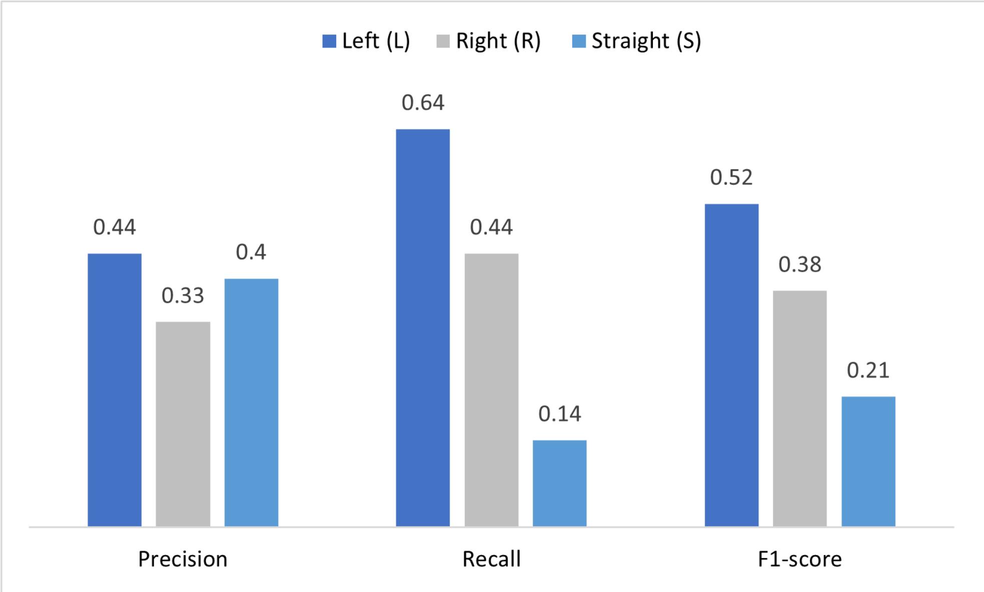


Figure 24. Performance of Perceptron

By interpreting the result of table 6, it is noticed, Figure 25, comparing the Precision score of four models based on the direction left, RF outperformed other models with 0.81 score which is quite satisfactory. Followed by Perceptron with 0.44 score and Naïve Bayes with 0.25 score. SVM perform the worst among all these machine learning models with 0.19 precision score for left direction. For direction right, the best Precision score also achieved by RF model with 0.65. In this case, Naïve Bayes with 0.50 gained the second-best Precision score. SVM and Perceptron performed quite similar achieving 0.37 and 0.33 score, respectively. Random Forest acquired the best Precision score again with 0.92 in the case of direction straight. Second-best performance is achieved by SVM in this case with score of 0.70. Naïve Bayes and Perceptron performed same with 0.40 Precision score.

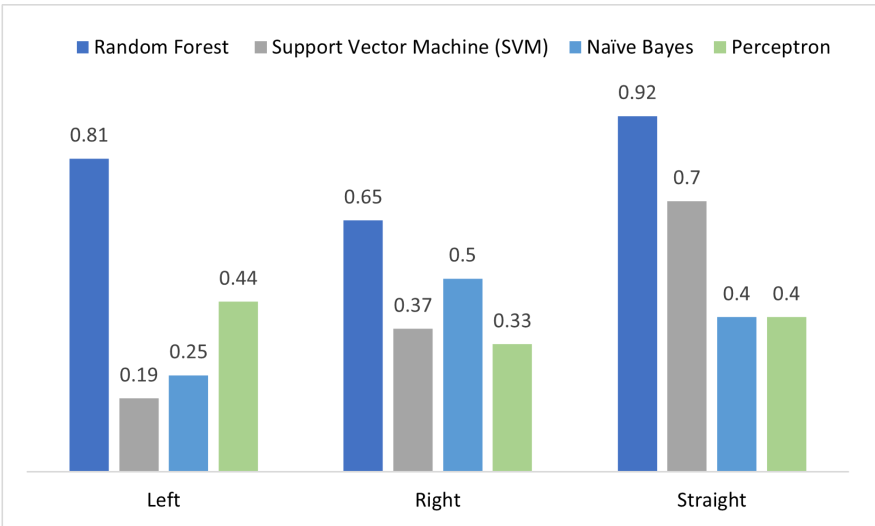


Figure 25. Comparison of Precision score

In terms of Recall score, Figure 26, for the direction of left, RF and Perceptron performed quite similar with the score of 0.68 and 0.64, correspondingly. On the other hand, the performance of SVM and Naïve Bayes is very low in this case. SVM gained only 0.12 recall score, and Naïve Bayes achieved only 0.08 score. For the direction right, best recall score acquired by RF with 0.88, in contrast, Naïve Bayes performed worst in this case with 0.28 recall score. SVM recall score is quite satisfactory in case of direction right which is 0.64, followed by Perceptron with 0.44. The best recall score, for the direction straight, is gained by model Naïve Bayes with 0.79, closely followed by RF with score of 0.76. The performance of SVM is reasonable that scored 0.48, however Perceptron achieved very low score, 0.14, in case of recall for straight direction.

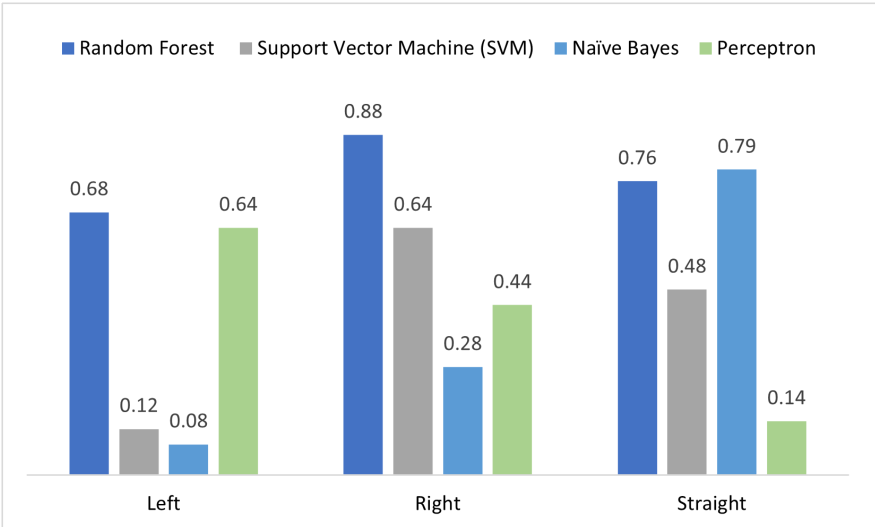


Figure 26. Comparison of Recall score

Lastly, if the f1-score is compared among four machine learning models, Figure 27, it is addressed that for the direction of left, RF performed best with 0.74 score, whereas SVM and Naïve Bayes achieved very low score, 0.15 and 0.12, respectively. The performance of Perceptron model is quite satisfactory, scored 0.52, in this case. For direction right, RF again achieved best f1-score with 0.75. The second-best performance is gained by SVM with 0.47 score. Followed by Perceptron and Naïve Bayes models, in this case, scored 0.38 and 0.36, respectively. In the case of straight direction, the best f1-score is achieved by random forest model with 0.83. SVM and Naïve Bayes also performed quite well, respectively scored 0.57 and 0.53. The worst performance for f1-score is done by Perceptron with only 0.21, in the case of straight direction.

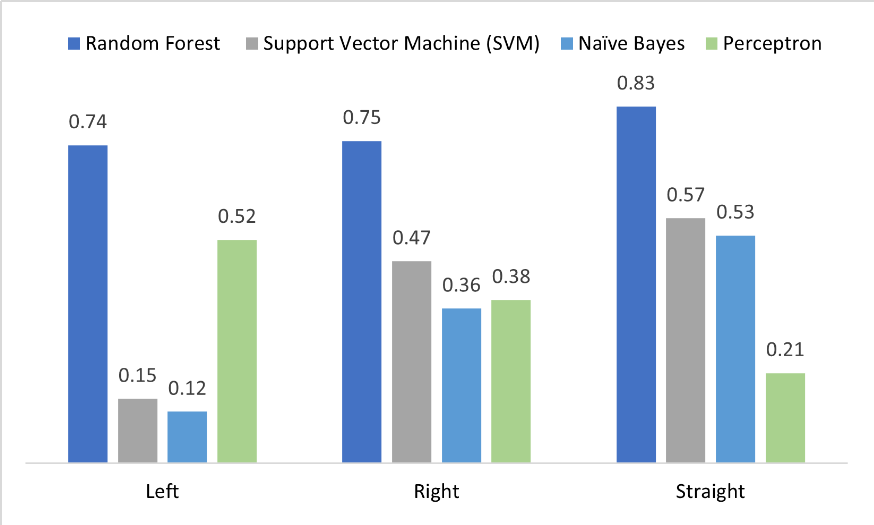


Figure 27. Comparison of F1-score

In order to determine the significant feature, *feature_importance_* property of Random Forest implementation for sklearn is applied. The result is given in table 7.

Table 7. Feature Importance Scores

Feature	**Importance Score**
Autocorrelation	0.11311929
Kurtosis	0.08534978
IQR	0.07441599
Skewness	0.07283231
Max	0.07179622
RMS	0.06497776
50%	0.06301705
Integral	0.06195987
Range	0.06000463
25%	0.05938238
STD	0.05556102
75%	0.05532057
Median	0.05481029
Variance	0.05480388
Mean	0.05264896

Result shows autocorrelation feature contribute most to the decision, followed by Kurtosis, IQR, and Skewness, correspondingly. Other features such as Mean, Variance, Median contribute least, respectively, to the decision.

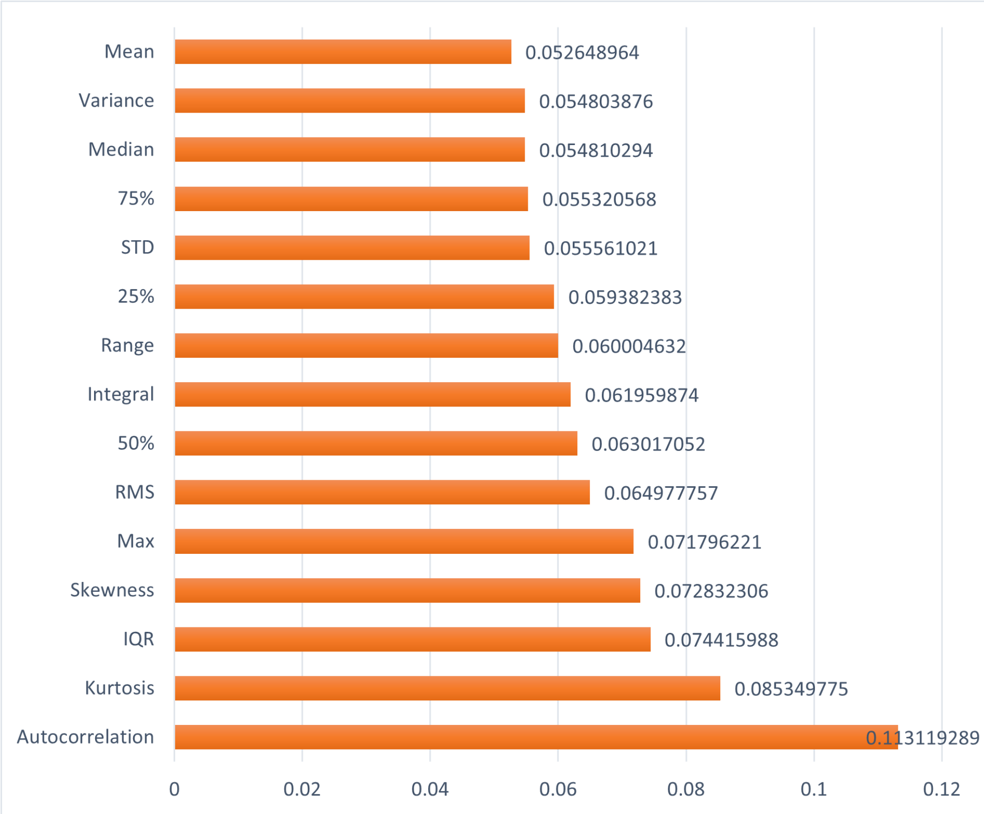


Figure 28. Feature Importance Scores

6.1. Answer to the Research Questions

This section focuses on discussing the answer to research questions mentioned in beginning of this thesis.

(RQ1) How accurately the head movement can be detected using gyroscope data collected through in-ear wearables?

From the result, it can be observed that the performance of machine learning models that have been applied in this thesis aiming to detect the head movement is not promising. Comparing all the models, Random Forest performs better than others, and it is able to detect the head movement with approximately 77% accuracy. The accuracy rate of other three models such as Support Vector Machine, Naïve Bayes, and Perceptron is close to each other, where they detect head movement with about 42%, 40%, and 39% accuracy, respectively. Considering the result, it can be mentioned that the of goal of detecting head movement applying machine learning models using gyroscope data collected via in-ear wearables is achieved, although the accuracy rate of machine learning models is not up to expectation.

Besides, the performance of these models is also evaluated by some other evaluation metrics like Precision, Recall, and F1-score. This evaluation result verifies that using these machine learning models, different head direction such left, right, or straight can be detected.

(RQ2) Determine whether the head movement detection is able to enhance the human capability in given scenario such as biking or driving?

Throughout the literature review, it is learned that if head movement detection can be achieved accurately in real-time manner, it can be applied potentially in various field of applications. Such as number of studies carried out aiming to identify drivers' head position in real-time manner while they drive. The objective of such studies is to help drivers or make them aware about the real-life circumstance during their driving. For example, drivers can be notified if their heads are not in right direction while they drive, or if they feel drowsiness, or somehow they become distracted. These types of attempts have significant impact on improving the driving quality, meaning enhance human capability which certainly help to reduce number of road accidents.

The result achieved in this thesis indicates that these machine learning models are capable of detecting different head direction with satisfactory accuracy, hence it can be mentioned that using these models applications can be developed in-real time manner in future. However, human capability and its enhancement were not possible to study in this thesis due to the pandemic. It remains as future work to test the effectiveness of these models in real world driving scenario. Therefore, relying on literature it can be stated that head movement detection is able to enhance human capability in given scenario such as biking or driving.

7. DISCUSSION

Data collection challenges: Although data collection was not part of this thesis work, it is acknowledged that data was collected in lab environment. Due to Covid-19 situation, it was not allowed to meet with people and collect data in real life environment. Participants were given eSense device and were instructed how to use it. According to guideline, they wore eSense device and moved their heads in different direction – left, right and straight – for a specific time frame in order to generate accelerometer and gyroscope data. The task of moving head was performed by participants while they were sitting in a chair. Data was saved locally using an Android mobile application. Because of data collection was done in lab environment, it is believed the accelerometer data was not sufficient to utilize in this thesis. If data could be collected in real life situation such as participant is wearing eSense device while driving or biking, data could have been more authentic to verify the performance of machine learning models as using accelerometer and gyroscope data together bring out best performance in predictive assessment.

Challenges in features extraction: The first challenge that has been encountered was finding appropriate features matching with the purpose of this study. Features were required to be extracted aiming to develop different machine learning models to detect the head direction. In order to achieve this objective, huge number of similar literature was studied which certainly helped to gain adequate knowledge about others' work. After giving so much effort, the goal of finding suitable features was achieved, features were chosen for this study based on an similar analysis of accelerometer data conducted by Hemminki at al. [92]. Another challenge in feature extraction part was file segmentation. Each csv file had accelerometer and gyroscope data for three labeled direction – straight, left, and right, and there were 30 files having same kind of data. In order to extract feature, each file was required to separated in three different csv files based on direction. This process was fairly challenging as it was time consuming as well as had to apply same algorithms for all 90 csv files to gain the feature extraction values.

Limitations: One of the important elements of predictive analysis is data, hence data collection process plays a crucial role in such study. Consequently, it can be stated that the key limitation of this study is data collection process happened due to Covid-19 situation. With real life data, using both accelerometer and gyroscope data, machine learning models are expected to perform better.

Future Work: The goal of this thesis is fairly achieved, however, in future the aim should be set on collecting data in real life situation, subsequently the head movement using machine learning model can be detected more accurately. As it is evident now head movement can be detected using gyroscope data generated via eSene device, thus in future aim should be set on developing algorithms to create various applications using eSense that can detect head movement in real-time manner focusing to enhance human capability in real life situations.

8. CONCLUSION

This thesis is motivated by the study carried out in [19] by Ferlini et al. where authors also used eSense device to track head movement during different activities such as standing in silence, chewing, and speaking. Authors focused to evaluate the performance of eSense in terms of measuring up to how much degrees the head movement can be detected. Result shows inertial sensors in eSense achieve result precise up to few degrees, however for longer head movements, the estimation declines. With the motivation from this paper, in this thesis the goal is set to apply various machine learning models that can detect head movement. To achieve this goal, eSense device is utilized to collect gyroscope data. Features are extracted using this gyroscope data. Subsequently, extracted features are used to apply machine learning models. Considering the outcome of this thesis, it can be concluded that different head direction can be detected applying machine learning models employing gyroscope data collected through in-ear wearable, eSense.

Five participants were involved in data collection process. Data collection was done by others, it was not part of this thesis work. Each participant performed the task of moving head in three direction – straight, left, and right for six times, hence, there was 30 files all together in csv format in dataset. Participants were wearing eSesne device that has built-in inertial sensors such as accelerometer and gyroscope. The data was saved locally using an Android application. All the files contain readings about three axis accelerometer, three axis gyroscopes, as well as labeled direction of left, right and straight. For features extraction, each file was segmented into three csv files based on direction. After segmenting 30 files into 90 csv files, same algorithm was used to extract the features from each file. Subsequently, all feature files were merged to one single csv file.

Based on these extracted features, four machine learning models are applied. Four machine learning models are Random Forest, Support Vector Machine, Naïve Bayes, and Perceptron. In order to evaluate the performance of these models, various evaluation metrics like Accuracy, Precision, Recall, and F1 score are applied. The result indicates the performance of all these machine learning models are not promising, however, the goal of detecting head movement is achieved. Comparing all the models, Random Forest performs better than others with about 77% accuracy. The performance of other three models, Support Vector Machine, Naïve Bayes, and Perceptron are not up to expectation as well. Support Vector Machine achieves approximately 41% accuracy. For Naïve Bayes and Perceptron, the accuracy scores are about 40% and 39%, respectively. However, other evaluation metrics also justify that all machine learning models are able to detect different head direction. Studying the information gathered through literature review and analyzing the result of this thesis, it can be outlined that the contribution of this work in the field of head movement detection technology is considerable focusing on future work of developing real-time application to enhance human capability in various real-life situations. Hence lastly it can be concluded that the objectives of this thesis are fairly achieved.

9. REFERENCES

[1] Perwej Y., Haq K., Parwej F., Mumdouh M. & Hassan M. (2019) The internet of things (iot) and its application domains. International Journal of Computer Applications 975, p. 182.

[2] Zhu N., Diethe T., Camplani M., Tao L., Burrows A., Twomey N., Kaleshi D., Mirmehdi M., Flach P. & Craddock I. (2015) Bridging e-health and the internet of things: The sphere project. IEEE Intelligent Systems 30, pp. 39–46.

[3] Al-Rahayfeh A. & Faezipour M. (2013) Eye tracking and head movement detection: A state-of-art survey. IEEE journal of translational engineering in health and medicine 1, pp. 2100212–2100212.

[4] Sasou A. (2009) Acoustic head orientation estimation applied to powered wheelchair control. In: 2009 Second International Conference on Robot Communication and Coordination, IEEE, pp. 1–6.

[5] King L., Nguyen H. & Taylor P. (2006) Hands-free head-movement gesture recognition using artificial neural networks and the magnified gradient function. In: 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference, IEEE, pp. 2063–2066.

[6] Xie D., Dang L. & Tong R. (2012) Video based head detection and tracking surveillance system. In: 2012 9th International Conference on Fuzzy Systems and Knowledge Discovery, IEEE, pp. 2832–2836.

[7] Liu K., Luo Y., Gyomei T. & Yang S. (2008) Attention recognition of drivers based on head pose estimation. In: 2008 IEEE Vehicle Power and Propulsion Conference, IEEE, pp. 1–5.

[8] Lee D., Oh S., Heo S. & Hahn M. (2008) Drowsy driving detection based on the driver's head movement using infrared sensors. In: 2008 Second International Symposium on Universal Communication, IEEE, pp. 231–236.

[9] Nguyen S.T., Nguyen H.T., Taylor P.B. & Middleton J. (2006) Improved head direction command classification using an optimised bayesian neural network. In: 2006 International Conference of the IEEE Engineering in Medicine and Biology Society, IEEE, pp. 5679–5682.

[10] Manogna S., Vaishnavi S. & Geethanjali B. (2010) Head movement based assist system for physically challenged. In: 2010 4th International Conference on Bioinformatics and Biomedical Engineering, IEEE, pp. 1–4.

[11] Kim S., Park M., Anumas S. & Yoo J. (2010) Head mouse system based on gyro-and opto-sensors. In: 2010 3rd International Conference on Biomedical Engineering and Informatics, vol. 4, IEEE, vol. 4, pp. 1503–1506.

[12] Murphy-Chutorian E. & Trivedi M.M. (2010) Head pose estimation and augmented reality tracking: An integrated system and evaluation for monitoring

driver awareness. IEEE Transactions on intelligent transportation systems 11, pp. 300–311.

[13] Kupetz D., Wentzell S. & BuSha B. (2010) Head motion controlled power wheelchair. In: Proceedings of the 2010 IEEE 36th Annual Northeast Bioengineering Conference (NEBEC), IEEE, pp. 1–2.

[14] Francois A.R. (2004) Real-time multi-resolution blob tracking. Tech. rep., UNIVERSITY OF SOUTHERN CALIFORNIA LOS ANGELES INST FOR ROBOTICS AND . . . .

[15] Jian-zheng L. & Zheng Z. (2011) Head movement recognition based on lk algorithm and gentleboost. In: The 7th International Conference on Networked Computing and Advanced Information Management, IEEE, pp. 232–236.

[16] Berjón R., Mateos M., Barriuso A.L., Muriel I. & Villarrubia G. (2011) Alternative human-machine interface system for powered wheelchairs. In: 2011 IEEE 1st International Conference on Serious Games and Applications for Health (SeGAH), IEEE, pp. 1–5.

[17] Zhao Z., Wang Y. & Fu S. (2012) Head movement recognition based on lucas-kanade algorithm. In: 2012 International Conference on Computer Science and Service System, IEEE, pp. 2303–2306.

[18] Kawsar F., Min C., Mathur A. & Montanari A. (2018) Earables for personal-scale behavior analytics. IEEE Pervasive Computing 17, pp. 83–89.

[19] Ferlini A., Montanari A., Mascolo C. & Harle R. (2019) Head motion tracking through in-ear wearables. In: Proceedings of the 1st International Workshop on Earable Computing, pp. 8–13.

[20] Favre-Felix A., Graversen C., Hietkamp R.K., Dau T. & Lunner T. (2018) Improving speech intelligibility by hearing aid eye-gaze steering: Conditions with head fixated in a multitalker environment. Trends in hearing 22, p. 2331216518814388.

[21] Fernandez P. (2014) Wearable technology: beyond augmented reality. Library Hi Tech News .

[22] Park S. & Jayaraman S. (2021) Wearables: Fundamentals, advancements, and a roadmap for the future. In: Wearable sensors, Elsevier, pp. 3–27.

[23] Qiu H., Wang X. & Xie F. (2017) A survey on smart wearables in the application of fitness. In: 2017 IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech), IEEE, pp. 303–307.

[24] Yang Z., Wang Z., Zhang J., Huang C. & Zhang Q. (2015) Wearables can afford: Light-weight indoor positioning with visible light. In: Proceedings of the 13th annual international conference on mobile systems, applications, and services, pp. 317–330.

[25] Lee Y.H. & Medioni G. (2016) Rgb-d camera based wearable navigation system for the visually impaired. Computer Vision and Image Understanding 149, pp. 3–20.

[26] Anzaldo D. (2015) Wearable sports technology-market landscape and compute soc trends. In: 2015 International SoC Design Conference (ISOCC), IEEE, pp. 217–218.

[27] Seneviratne S., Hu Y., Nguyen T., Lan G., Khalifa S., Thilakarathna K., Hassan M. & Seneviratne A. (2017) A survey of wearable devices and challenges. IEEE Communications Surveys & Tutorials 19, pp. 2573–2620.

[28] Xu W., Shen Y., Zhang Y., Bergmann N. & Hu W. (2017) Gait-watch: A context-aware authentication system for smart watch based on gait recognition. In: Proceedings of the Second International Conference on Internet-of-Things Design and Implementation, pp. 59–70.

[29] Gruenerbl A., Pirkl G., Monger E., Gobbi M. & Lukowicz P. (2015) Smart-watch life saver: smart-watch interactive-feedback system for improving bystander cpr. In: Proceedings of the 2015 ACM International Symposium on Wearable Computers, pp. 19–26.

[30] Xu C., Pathak P.H. & Mohapatra P. (2015) Finger-writing with smartwatch: A case for finger and hand gesture recognition using smartwatch. In: Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications, pp. 9–14.

[31] Cornelius C., Peterson R., Skinner J., Halter R. & Kotz D. (2014) A wearable system that knows who wears it. In: Proceedings of the 12th annual international conference on Mobile systems, applications, and services, pp. 55–67.

[32] Parate A., Chiu M.C., Chadowitz C., Ganesan D. & Kalogerakis E. (2014) Risq: Recognizing smoking gestures with inertial sensors on a wristband. In: Proceedings of the 12th annual international conference on Mobile systems, applications, and services, pp. 149–161.

[33] Anliker U., Ward J.A., Lukowicz P., Troster G., Dolveck F., Baer M., Keita F., Schenker E.B., Catarsi F., Coluccini L. et al. (2004) Amon: a wearable multiparameter medical monitoring and alert system. IEEE Transactions on information technology in Biomedicine 8, pp. 415–427.

[34] Habetha J. (2006) The myheart project-fighting cardiovascular diseases by prevention and early diagnosis. In: 2006 International Conference of the IEEE Engineering in Medicine and Biology Society, IEEE, pp. 6746–6749.

[35] Luprano J., Solà J., Dasen S., Koller J.M. & Chételat O. (2006) Combination of body sensor networks and on-body signal processing algorithms: the practical case of myheart project. In: International Workshop on Wearable and Implantable Body Sensor Networks (BSN'06), IEEE, pp. 4–pp.

[36] Pacelli M., Loriga G., Taccini N. & Paradiso R. (2006) Sensing fabrics for monitoring physiological and biomechanical variables: E-textile solutions. In: 2006 3rd IEEE/EMBS International Summer School on Medical Devices and Biosensors, IEEE, pp. 1–4.

[37] Pandian P., Mohanavelu K., Safeer K., Kotresh T., Shakunthala D., Gopal P. & Padaki V. (2008) Smart vest: Wearable multi-parameter remote physiological monitoring system. Medical engineering & physics 30, pp. 466–477.

[38] LeBoeuf S.F., Aumer M.E., Kraus W.E., Johnson J.L. & Duscha B. (2014) Earbud-based sensor for the assessment of energy expenditure, heart rate, and vo2max. Medicine and science in sports and exercise 46, p. 1046.

[39] Bedri A., Li R., Haynes M., Kosaraju R.P., Grover I., Prioleau T., Beh M.Y., Goel M., Starner T. & Abowd G. (2017) Earbit: using wearable sensors to detect eating episodes in unconstrained environments. Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies 1, pp. 1–20.

[40] Amft O., Stäger M., Lukowicz P. & Tröster G. (2005) Analysis of chewing sounds for dietary monitoring. In: International Conference on Ubiquitous Computing, Springer, pp. 56–72.

[41] Zheng W.L., Zhu J.Y. & Lu B.L. (2017) Identifying stable patterns over time for emotion recognition from eeg. IEEE Transactions on Affective Computing 10, pp. 417–429.

[42] Athavipach C., Pan-Ngum S. & Israsena P. (2019) A wearable in-ear eeg device for emotion monitoring. Sensors 19, p. 4014.

[43] Matthies D.J. (2013) Inear biofeedcontroller: a headset for hands-free and eyes-free interaction with mobile devices. In: CHI'13 Extended Abstracts on Human Factors in Computing Systems, pp. 1293–1298.

[44] Matthies D.J., Strecker B.A. & Urban B. (2017) Earfieldsensing: A novel in-ear electric field sensing to enrich wearable gesture input through facial expressions. In: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, pp. 1911–1922.

[45] Jeong D.H. & Jeong J. (2020) In-ear eeg based attention state classification using echo state network. Brain sciences 10, p. 321.

[46] Goverdovsky V., Von Rosenberg W., Nakamura T., Looney D., Sharp D.J., Papavassiliou C., Morrell M.J. & Mandic D.P. (2017) Hearables: Multimodal physiological in-ear sensing. Scientific reports 7, pp. 1–10.

[47] Lee S., Min C., Montanari A., Mathur A., Chang Y., Song J. & Kawsar F. (2019) Automatic smile and frown recognition with kinetic earables. In: Proceedings of the 10th Augmented Human International Conference 2019, pp. 1–4.

[48] Katayama S., Mathur A., Okoshi T., Nakazawa J. & Kawsar F. (2019) Situation-aware conversational agent with kinetic earables. In: Proceedings of the 17th Annual International Conference on Mobile Systems, Applications, and Services, pp. 657–658.

[49] Ishii S., Yokokubo A., Luimula M. & Lopez G. (2021) Exersense: Physical exercise recognition and counting algorithm from wearables robust to positioning. Sensors 21, p. 91.

[50] Islam M.S., Hossain T., Ahad M.A.R. & Inoue S. (2021) Exploring human activities using esense earable device. In: Activity and Behavior Computing, Springer, pp. 169–185.

[51] Purabi S.A., Rashed R., Islam M., Uddin N., Naznin M. & Islam A.A.A. (2019) As you are, so shall you move your head: a system-level analysis between head movements and corresponding traits and emotions. In: Proceedings of the 6th International Conference on Networking, Systems and Security, pp. 3–11.

[52] Kawsar F., Min C., Mathur A., Montanari A., Acer U.G. & Van den Broeck M. (2018) Demo abstract: esense-open earable platform for human sensing. In: 16th ACM Conference on Embedded Networked Sensor Systems, SENSYS 2018, Association for Computing Machinery (ACM), pp. 371–372.

[53] Arriba-Pérez D., Caeiro-Rodríguez M., Santos-Gago J.M. et al. (2016) Collection and processing of data from wrist wearable devices in heterogeneous and multiple-user scenarios. Sensors 16, p. 1538.

[54] Shen S., Gowda M. & Roy Choudhury R. (2018) Closing the gaps in inertial motion tracking. In: Proceedings of the 24th Annual International Conference on Mobile Computing and Networking, pp. 429–444.

[55] Tolle H. & Arai K. (2016) Design of head movement controller system (hemocs) for control mobile application through head pose movement detection. International Journal of Interactive Mobile Technologies (iJIM) 10, pp. 24–28.

[56] Varma A.R., Arote S.V., Bharti C. & Singh K. (2012) Accident prevention using eye blinking and head movement. Emerging Trends in Computer Science and Information Technology (ETCSIT2012). Proceedings published in International Journal of Computer Applications (IJCA) , pp. 18–22.

[57] Kim E., Helal S. & Cook D. (2009) Human activity recognition and pattern discovery. IEEE pervasive computing 9, pp. 48–53.

[58] Madabhushi A. & Aggarwal J. (2000) Using head movement to recognize activity. In: Proceedings 15th International Conference on Pattern Recognition. ICPR-2000, vol. 4, IEEE, vol. 4, pp. 698–701.

[59] Rechy-Ramirez E.J., Hu H. & McDonald-Maier K. (2012) Head movements based control of an intelligent wheelchair in an indoor environment. In: 2012 IEEE International Conference on Robotics and Biomimetics (ROBIO), IEEE, pp. 1464–1469.

[60] Gray J., Jia P., Hu H.H., Lu T. & Yuan K. (2007) Head gesture recognition for hands-free control of an intelligent wheelchair. Industrial Robot: An International Journal .

[61] Craig D.A. & Nguyen H.T. (2006) Wireless real-time head movement system using a personal digital assistant (pda) for control of a power wheelchair. In: 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference, IEEE, pp. 772–775.

[62] Katevas N.I., Sgouros N.M., Tzafestas S.G., Papakonstantinou G., Beattie P., Bishop J.M., Tsanakas P. & Koutsouris D. (1997) The autonomous mobile robot senario: a sensor aided intelligent navigation system for powered wheelchairs. IEEE Robotics & Automation Magazine 4, pp. 60–70.

[63] Bourhis G., Horn O., Habert O. & Pruski A. (2001) An autonomous vehicle for people with motor disabilities. IEEE Robotics & Automation Magazine 8, pp. 20–28.

[64] Lankenau A. & Rofer T. (2001) A versatile and safe mobility assistant. IEEE Robotics & Automation Magazine 8, pp. 29–37.

[65] Mazo M. (2001) An integral system for assisted mobility [automated wheelchair]. IEEE Robotics & Automation Magazine 8, pp. 46–56.

[66] Yanco H.A. (1998) Wheelesley: A robotic wheelchair system: Indoor navigation and user interface. In: Assistive technology and artificial intelligence, Springer, pp. 256–268.

[67] Kuo C.H. & Chen H.H. (2006) Human-oriented design of autonomous navigation assisted robotic wheelchair for indoor environments. In: 2006 IEEE International Conference on Mechatronics, IEEE, pp. 230–235.

[68] Pajkanović A. & Dokić B. (2013) Wheelchair control by head motion. Serbian Journal of electrical engineering 10, pp. 135–151.

[69] Elleuch H., Wali A. & Alimi A.M. (2014) Smart tablet monitoring by a real-time head movement and eye gestures recognition system. In: 2014 International Conference on Future Internet of Things and Cloud, IEEE, pp. 393–398.

[70] Viola P. & Jones M.J. (2004) Robust real-time face detection. International journal of computer vision 57, pp. 137–154.

[71] Fanani L., Brata A.H. & Dewi DP R.P. (2019) An interactive mobile technology to improve the usability of exam application for disabled student. In: Proceedings of the 2019 3rd International Conference on Education and Multimedia Technology, pp. 302–306.

[72] Hadar U., Steiner T.J., Grant E.C. & Clifford Rose F. (1983) Head movement correlates of juncture and stress at sentence level. Language and speech 26, pp. 117–129.

[73] Kendon A. (1975) Some functions of the face in a kissing round .

[74] McClave E.Z. (2000) Linguistic functions of head movements in the context of speech. Journal of pragmatics 32, pp. 855–878.

[75] Guan D., Ma T., Yuan W., Lee Y.K. & Jehad Sarkar A. (2011) Review of sensor-based activity recognition systems. IETE Technical Review 28, pp. 418–433.

[76] Huang L., Zhou Z. & Liu R. (2014) Research on interaction-oriented gesture recognition. Sensors & Transducers 163, p. 316.

[77] Muthohar M.F., Nugraha I.G.D. & Choi D. Browse article .

[78] Jung Y. & Cha B. (2010) Gesture recognition based on motion inertial sensors for ubiquitous interactive game contents. IETE Technical review 27, pp. 158–166.

[79] Priandani N.D., Tolle H. & Utaminingrum F. (2017) Real time advanced head move-ment recognition for application controller based on android internal gyroscope sen-sor. Int. J. Adv. Soft Comput. Its Appl 9.

[80] Yunardi R.T., Dina N.Z., Agustin E.I. & Firdaus A.A. (2020) Visual and gyroscope sensor for head movement controller system on meal-assistance application. Majlesi Journal of Electrical Engineering 14, pp. 39–44.

[81] Tadano K. & Kawashima K. (2015) A pneumatic laparoscope holder controlled by head movement. The International Journal of Medical Robotics and Computer Assisted Surgery 11, pp. 331–340.

[82] Jain A. & Kanhangad V. (2017) Human activity classification in smartphones using accelerometer and gyroscope sensors. IEEE Sensors Journal 18, pp. 1169–1177.

[83] Patel N., Dubey T., Tomar S.S. & Sharma M.K. Human activity identification using smartphone data .

[84] Muset B. & Emerich S. (2012) Distance measuring using accelerometer and gyroscope sensors. Carpathian Journal of Electronic and Computer Engineering 5, p. 83.

[85] Bonnet V., Mazza C., McCamley J. & Cappozzo A. (2013) Use of weighted fourier linear combiner filters to estimate lower trunk 3d orientation from gyroscope sensors data. Journal of neuroengineering and rehabilitation 10, pp. 1–8.

[86] Wibisono W., Arifin D.N., Pratomo B.A., Ahmad T. & Ijtihadie R.M. (2013) Falls detection and notification system using tri-axial accelerometer and gyroscope sensors of a smartphone. In: 2013 Conference on Technologies and Applications of Artificial Intelligence, IEEE, pp. 382–385.

[87] of Violence W.H.O.D., Prevention I., Violence W.H.O., Prevention I. & Organization W.H. (2009) Global status report on road safety: time for action. World Health Organization.

[88] Allouch A., Koubâa A., Abbes T. & Ammar A. (2017) Roadsense: Smartphone application to estimate road conditions using accelerometer and gyroscope. IEEE Sensors Journal 17, pp. 4231–4238.

[89] Barshan B. & Durrant-Whyte H.F. (1995) Evaluation of a solid-state gyroscope for robotics applications. IEEE Transactions on Instrumentation and measurement 44, pp. 61–67.

[90] Nokia-Bell-Labs (2019), Microsoft word - esense-user-documentation.docx. https://www.esense.io/share/eSense-User-Documentation.pdf. (Accessed on 05/25/2021).

[91] Nokia-Bell-Labs (2019), Microsoft word - esense-visualiser-documentation.docx. https://www.esense.io/share/eSense-Visualiser-Documentation.pdf. (Accessed on 05/26/2021).

[92] Hemminki S., Nurmi P. & Tarkoma S. (2013) Accelerometer-based transportation mode detection on smartphones. In: Proceedings of the 11th ACM conference on embedded networked sensor systems, pp. 1–14.

[93] Wikipedia (2021), Mean — Wikipedia, the free encyclopedia. http://en.wikipedia.org/w/index.php?title=Mean&oldid=1024415554. [Online; accessed 26-May-2021].

[94] How to calculate the column variance of a dataframe in python pandas? | finxter. https://blog.finxter.com/how-to-calculate-the-column-variance-of-a-dataframe-in-python-pandas/. (Accessed on 05/26/2021).

[95] Standard deviation - wikipedia. https://en.wikipedia.org/wiki/Standard_deviation. (Accessed on 05/26/2021).

[96] Standard deviation and variance. https://www.mathsisfun.com/data/standard-deviation.html. (Accessed on 05/26/2021).

[97] Percentile - wikipedia. https://en.wikipedia.org/wiki/Percentile. (Accessed on 05/26/2021).

[98] Percentiles. https://www.mathsisfun.com/data/percentiles.html. (Accessed on 05/26/2021).

[99] Nedarc - min, max, and range. https://www.nedarc.org/statisticalhelp/basicStatistics/minAndMax.html. (Accessed on 05/26/2021).

[100] Median - wikipedia. https://en.wikipedia.org/wiki/Median. (Accessed on 05/26/2021).

[101] 1.3.5.11. measures of skewness and kurtosis. https://www.itl.nist.gov/div898/handbook/eda/section3/eda35b.htm. (Accessed on 05/26/2021).

[102] Skewness - wikipedia. https://en.wikipedia.org/wiki/Skewness. (Accessed on 05/26/2021).

[103] Kurtosis - wikipedia. https://en.wikipedia.org/wiki/Kurtosis. (Accessed on 05/26/2021).

[104] Interquartile range | understand, calculate & visualize iqr. https://www.scribbr.com/statistics/interquartile-range/. (Accessed on 05/26/2021).

[105] The root mean square. http://www.analytictech.com/mb313/rootmean.htm. (Accessed on 05/26/2021).

[106] Root mean square - wikipedia. https://en.wikipedia.org/wiki/Root_mean_square. (Accessed on 05/26/2021).

[107] Integral - wikipedia. https://en.wikipedia.org/wiki/Integral. (Accessed on 05/26/2021).

[108] Autocorrelation - wikipedia. https://en.wikipedia.org/wiki/Autocorrelation. (Accessed on 05/26/2021).

[109] Murphy K.P. (2012) Machine learning: a probabilistic perspective. MIT press.

[110] El Naqa I. & Murphy M.J. (2015) What is machine learning? In: machine learning in radiation oncology, Springer, pp. 3–11.

[111] Chen G., Li S., Knibbs L.D., Hamm N.A., Cao W., Li T., Guo J., Ren H., Abramson M.J. & Guo Y. (2018) A machine learning method to estimate pm2. 5 concentrations across china with remote sensing, meteorological and land use information. Science of the Total Environment 636, pp. 52–60.

[112] Biau G. & Scornet E. (2016) A random forest guided tour. Test 25, pp. 197–227.

[113] Fawagreh K., Gaber M.M. & Elyan E. (2014) Random forests: from early developments to recent advancements. Systems Science & Control Engineering: An Open Access Journal 2, pp. 602–609.

[114] Jakkula V. (2006) Tutorial on support vector machine (svm). School of EECS, Washington State University 37.

[115] Patton M.Q. (2015) Qualitative research & evaluation methods: Integrating theory and practice .

[116] Makridakis S. (1993) Accuracy measures: theoretical and practical concerns. International journal of forecasting 9, pp. 527–529.

[117] Wikipedia (2021), Precision and recall — Wikipedia, the free encyclopedia. http://en.wikipedia.org/w/index.php?title=Precision%20and%20recall&oldid=1023690484. [Online; accessed 20-May-2021].

[118] How to calculate feature importance with python. https://machinelearningmastery.com/calculate-feature-importance-with-python/. (Accessed on 05/30/2021).