INTERNSHIP PROJECT REPORT

on

"Artificial Intelligence based Smart Knee Brace"

Under the guidance of

Prof. T. Kishore Kumar

Professor, Department of ECE

NIT Warangal

Submitted by

KODI VIDYA SAGAR *(2021BITE091)*

NATIONAL INSTITUTE OF TECHNOLOGY SRINAGAR
(An autonomous Institute of National Importance under the aegis of Ministry of Education, Govt. of India)
DEPARTMENT OF TRAINING & PLACEMENT
Tel/Fax: +91-9419226538,9419226574 Extn: 2130/31 Email: placements@nitsri.net
Hazratbal , Srinagar Jammu and Kashmir, 190006, INDIA

NO: NIT/T&P/INTP/2023-24/ITE/052
Date: 21-11-2023

TO WHOM IT MAY CONCERN

Subject: **Permission of Internship online/offline for student of NIT Srinagar.**

In – Plant/on–the –project internship/Practical Training is an important part of our engineering curriculum. This internship/training is regarded as a vital component of engineering education and is an indicator of extent of field experience, which is very essential for attaining excellence in the technical education. In this context, **Mr. /Ms. Kodi Vidya Sagar,** Enrolment No: **2021BITE091** pursuing B. Tech in INFORMATION TECHNOLOGY DEPARTMENT (2021-2025) in this Institute has completed his/her 4th semester of the degree (pursuing 5th semester) and is interested in 45 days internship in your esteemed organization.

It will be highly appreciated if your organization provides him/her a chance to get an exposure to some project related to him/her branch of engineering online/offline that is being carried out by your organization during winter vacation from 15th December 2023 to 15th February 2024.

We fervently hope that you will accede to our request and allow him/her to pursue him/her internship in your esteemed organization. The student has been advised to abide by the rules and regulation of your organization. Also, the student has to submit completion report and certificate in the training & placement department after completion of the internship, failing this his/her internship will be deemed incomplete.

Associate TPO (Internships)
Training and Placement
NIT Srinagar

Associate TPO (Internships)
Training & Placement Department
National Institute of Technology
Srinagar, J&K.

NATIONAL INSTITUTE OF TECHNOLOGY
WARANGAL-506004, TELANGANA (INDIA)

Department of Electronics & Communication Engineering

Dr. T. Kishore Kumar
M.Tech., Ph.D.,
Head CTL, Senior Professor of ECE, NITW,
Former Head of the Department of ECE, NITW,
Visiting Professor,
Telecommunications & IT,
School of Engineering & Technology,
Asian Institute of Technology
Bangkok, Thailand.

Phone: +918332969353
Email : *kishoret@nitw.ac.in,*
kishore@ait.asia,
kishorefr@gmail.com

TO WHOM IT MAY CONCERN

This is to certify that **Mr. Kodi Vidya Sagar (Reg No. 2021NITSGR0758),** student of III B.Tech., Department of Information Technology of NIT Srinagar. He has carried out his Winter Internship Under the supervision of **Prof.T.Kishore Kumar** in the Department of ECE, National Institute of Technology, Warangal. He has done his Internship work entitled "**Artificial Intelligence based Smart Knee Brace**". He has successfully completed Internship during the period of 15-12-2023 to 20-01-2024.

I, Appreciate his dedication and motivation towards work.

22/1/24.

Dr. T. Kishore Kumar
Professor,
Department of E.C.E
NIT Warangal

Dr. T. KISHORE KUMAR
M.Tech., Ph.D.
PROFESSOR
Electronics & Communication Engineering
National Institute of Technology
WARANGAL- 506004, T.S.

BONAFIED CERTIFICATE

This is to certify that this project report entitled **"Artificial Intelligence based Smart Knee Brace"** submitted to National Institute of Technology, Warangal and National Institute of Technology, Srinagar, is a bonafide record of work done by **"Kodi Vidya Sagar"** under my supervision from **"15 Dec 2023"** to **"20 Jan 2024".**

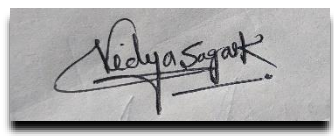
Supervisor
Prof. T. Kishore Kumar
Professor, Department of ECE
NIT Warangal

Place: Warangal

Date: 20 Jan 2021

DECLARATION

This is to declare that this report has been written by me. No part of the report is plagiarized from other sources. All information included from other sources have been duly acknowledged. I aver that if any part of the report is found to be plagiarized, I shall take full responsibility for it.



Kodi Vidya Sagar

2021BITE091

Place: Warangal

Date: 20 Jan 2024

ACKNOWLEDGEMENT

Completion of this project and thesis would not have been possible without the help of many people, to whom we are very thankful.

We would like to express my sincere gratitude to our supervisor Dr. T. Kishore Kumar, Professor, Head of CTL, NIT Warangal. His constant motivation, guidance and support helped us a great deal to achieve this feat.

We also thank our Guide M. Manoj Kumar, PhD Scholar for his constant support in guiding us about the academic related work.

We wish a deep sense of gratitude and heartfelt thanks to management for providing excellent lab facilities and tools. Finally, we thank our seniors whose guidance helped us in this regard.

ABSTRACT

With a focus primarily on knee angles, this research presents a novel method for measuring body joint angles. Examining the more general consequences of joint problems—which are especially relevant for the elderly and people with spinal cord injuries (SCI)—the technology shown here has great potential to alleviate knee problems that these populations encounter. Due to aging or the aftermath of an injury, older people and those with SCI are more likely to experience joint issues, such as osteoarthritis and decreased mobility. Seeing how urgently customized solutions are needed, the built technology is being used in areas outside of convention. Understanding and treating age-related joint problems greatly benefits from its use in tracking and evaluating knee joint motions in the elderly. The wearable sensory unit's use of sophisticated components, particularly the MPU6050 gyroscope sensor and Raspberry Pi, is crucial. This integration not only increases the computing capacity of the system but also expands its possible uses, from knee joint issues in the context of aging to rehabilitation scenarios. In the future, this research will need to be carefully expanded upon and refined. Subsequent efforts will concentrate on adding more sensors and adding features that are adaptable in order to tackle the unique problems that older people encounter. Through iterative development, a comprehensive system that smoothly matches with the specific demands of the aging population will be created, contributing to improved healthcare strategies catered to the intricacies of joint-related disorders and aging. All things considered, this cutting-edge technology has the potential to significantly alter healthcare paradigms by providing targeted monitoring and intervention strategies for knee joint issues, ultimately leading to better quality of life for senior citizens and people with spinal cord injuries.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS & ABBREVIATIONS

SCI	- Spinal Cord Injury
MPU6050	- Motion Processing Unit
IMU	- Inertial Measurement Unit
TKA	- Total Knee Arthroplasty
VNC	- Virtual Network Computing
RAM	- Random Access Memory
ROM	- Read Only Memory
GPIO	- General Purpose Input Output
IOT	- Internet of Things
USB	- Universal Serial Bus
MEMS	- Micro Electro Mechanical System
DOF	- Degree of Freedom
CSV	- Comma Separated Values
SMTP	- Simple Mail Transfer Protocol
VCC	- Voltage Supply
GND	- Ground Supply
SCL	- Serial Clock
SDA	- Serial Data Pin
EDA	- Auxiliary I2C Clock Pin
ECL	- Auxiliary I2C Clock Pin
ADO	- Address Select/Serial Data Output Pin

1. INTRODUCTION

1.1 Background to the study

The knee joint is necessary for supporting the body's weight and allowing the lower leg to move in relation to the thigh. Walking, running, sitting, and standing all need complicated knee actions. The knee, being a hinge synovial joint, allows for flexion and extension, which are necessary for a number of movements. Range of motion, or the maximum amount of mobility available in joints, has a significant impact on quality of life. Flexion and extension account for the majority of normal knee joint mobility, which is the primary range of motion.

Conversely, there are several ways in which the human knee can get damaged, such as direct trauma, overuse, and hyperextension (which includes varas and valgus components). These variables can lead to conditions like osteoarthritis, especially from knee injuries, as well as stroke and spinal cord injury (SCI) paralysis outcomes. Daily functioning may be significantly hampered by such traumas and neurological disorders. In order for people who suffer from these issues to regain their functionality, rehabilitation therapies are required. Because it ensures a full recovery, this restorative approach is crucial for individuals recovering from knee replacement surgery. This means that in order to help patients achieve their full potential again, rehabilitation plays a critical role in reducing the limitations brought on by neurological disorders, trauma, and surgical treatments.

We're tackling a wide range of prevalent and important joint health issues in society. Numerous people, especially those with spinal cord injuries (SCI) and the elderly, deal with musculoskeletal problems, which can cause problems including knee pain and restricted movement. Our project's main goal is to assist those who have common joint issues, particularly knee issues. Our goal is to provide consumers with a simple tool to monitor their joint health and identify any problems early on. With this strategy, we intend to make it simpler for everyone to maintain their health and take care of their joints.

1.2 Problem statement

“A Compassionate Journey into Joint Health Innovation”

A large segment of the population in today's society, particularly the elderly and those with spinal cord injuries, suffers from musculoskeletal issues that cause knee discomfort and limited mobility. Inadequate rehabilitation results and postponed intervention are caused by the lack of an easy-to-use and proactive way to track joint health. To close this gap, our initiative suggests a tool that allows people to track their joint health, making it easier to spot problems early and improving the efficacy of rehabilitation techniques. The main issue is that accessible monitoring and early intervention can help improve joint health management for people with knee-related problems.

1.3 Aim of the study

Our task is to develop a wearable system for measuring the knee angle with respect to time using a Raspberry Pi interfaced with a gyroscope sensor (MPU6050).

1.4 Objectives of the study

1. To measure knee joint angles during various activities (Sitting).
2. To track knee flexion and extension range of motion.
3. To send measured data periodically via email for remote monitoring and analysis in an excel sheet in tabular form.
4. To categorize the person’s condition as either normal or abnormal.

2. LITERATURE SURVEY

Many techniques have been used to quantify knee joint movement, and the goniometer—which is shown in—has gained widespread use. Its accuracy, however, depends on precise alignment with the joint center and requires repositioning each time the targeted joint moves. A digital imaging approach for measuring knee joint mobility was introduced by Bennett et al [1]., however its time-consuming nature is a limitation. Compared to other methods, this one has more logistical issues because it requires a digital camera, a computer, and angle measurement software.

Stretch sensors are a useful tool for measuring knee range of motion, as proposed by Somruthai et al [2]., who highlighted their constant deformation in response to applied forces. However, prior to clinical implementation, it is crucial to validate the methodology's reliability under a variety of scenarios. An alternative method for monitoring knee joint movement made use of optical fiber bending sensors, but it still needs improvement because the expected result shows a little angular displacement—not much less than the usual rotation range for knee applications.

In response to the demand for a lightweight, space-unrestricted wearable rehabilitation system, inertial sensors, also known as inertial measurement units (IMUs), have become a promising option. Because of its affordability, sturdy design, and ease of use, IMUs have shown effectiveness in health-related applications for calculating acceleration, angular rate, and magnetic field vectors in a three-dimensional local coordinate system. For knee rehabilitation activities, wearable devices based on IMUs have been thoroughly investigated. These technologies have uses ranging from gait analysis approaches to monitoring postoperative gait anomalies and fall detection.

Regarding knee flexion monitoring, a method that was developed in makes use of two wearable IMU sensors that are affixed to the legs of patients. This allows for the monitoring of knee flexion to occur at home following Total Knee Arthroplasty (TKA) surgery. A technique for detecting knee joint angles during cycling using two IMU sensors was presented by Cordillet et al [3]. These developments highlight how adaptable IMU-based technologies are in a range of rehabilitation settings, providing a viable path for accurate and easily accessible knee joint movement monitoring and analysis.

In introducing our “Artificial Intelligence based Smart Knee Brace”, meticulous consideration has been given to a comprehensive review of all relevant reference papers, ensuring a thorough understanding of existing research and advancements in the field. Our approach prioritizes cost-effectiveness, aligning with the imperative to make innovative healthcare solutions accessible. By incorporating insights from established literature and focusing on efficiency in resource utilization, we aim to deliver a smart knee brace that not only reflects the latest technological developments but is also economically viable. This commitment to both academic research and practical affordability underscores our dedication to creating impactful solutions in the realm of healthcare technology. The MPU6050 Gyroscope sensor's single use streamlines setup, which makes it ideal for certain uses. The use of the Gyroscope sensor to assess several aspects of knee joint mobility is introduced in this study. The MPU6050 Gyroscope sensor, VNC Viewer, and a Raspberry Pi, which serves as the work's central processing unit, are among the essential parts used. An Ethernet cable is used to link the Raspberry Pi to VNC Viewer so that code may be implemented smoothly and real-time angular position and velocity data logging is possible. This creative setup increases the system's adaptability for a range of uses.

3. METHODOLOGY

In this section, the system, its components, and the operation of each subsystem will be explained. The design and development of the system also will be explained which includes the hardware and software part.

3.1 Hardware Description

Raspberry Pi 3

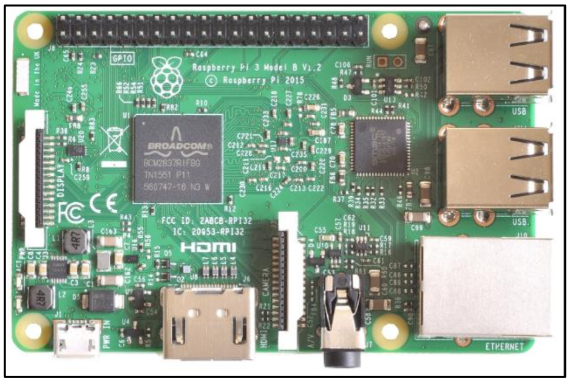


Fig 3.1 Raspberry Pi

Raspberry Pi 3 is a series of small, affordable, single board computers developed by Raspberry Pi foundation; a UK based charity organization. Similar to a traditional desktop CPU, the Raspberry Pi 3 functions as a mini computer, carrying out tasks that are typically handled by central processing unit (CPU). It is with built-in RAM and ROM components, the Raspberry Pi 3 operates much like a desktop computer, executing functions analogous to those performed by a CPU. It is a very cheap computer that runs Linux, but it also provides a set of GPIO (general purpose input/output) pins, allowing you to control electronic components for physical computing and explore the Internet of Things (IoT).

The Raspberry Pi 4 board has 40 pins that includes two- 5V Power Supply Pins, two- 3.3V Power Supply Pins, eight- Ground Pins, twenty-six- GPIOs pins and two-I2C IDS Pins. It has USB ports, Display ports, Camera ports, Audio Ports, Micro SD port. Additionally, it has Networks like Ethernet, WIFI and Bluetooth.

The Raspberry Pi 3 is more than just a cheap, small computer; it's meant to be a tool for learning and creativity through hands-on experience. Developed by the Raspberry Pi Foundation, its main objective is to offer a reasonably priced and easily obtainable platform that encourages users, educators, and hobbyists to delve into the fields of electronics, computer science, and programming. The Raspberry Pi acts as a spark for do-it-yourself endeavors, giving users the opportunity to develop prototypes, experiment with hardware, and learn the nuances of coding. Its adaptability and a strong developer community have led to its acceptance in a wide range of applications, from media centers and home automation systems to acting as the brains behind cutting-edge initiatives in industries like robotics, Internet of Things (IoT), and much beyond. As an instructional tool, the Raspberry Pi presents computational thinking and programming in a concrete and interesting way, which makes it a priceless asset for workshops and schools around the world. Basically, Raspberry Pi wants to lower the entry barriers into computers so that people of all ages and backgrounds can explore, be creative, and learn new things.

MPU6050 MODULE

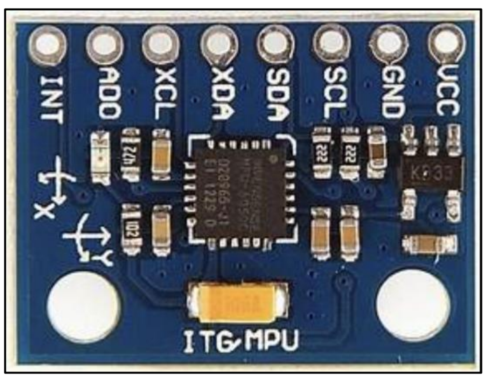


Fig 3.2 MPU6050 Module

MPU6050 is a multi-tasking sensor module based on MEMS (Micro Electro Mechanical Systems) architecture. It is a product by Asahi Kasei Microdevices Corporation. The MPU6050 is a 6-axial motion tracking device with a 3-dimensional Gyroscope, 3-Dimensional and 3-dimensional accelerometer embedded into an ultra-compact IC. The high-performance IC has 16-bit resolution analog to digital converters. The sensitive module also has VDDIO, an inbuilt temperature sensor, and an auxiliary I2C interface data transmission with non-inertial sensors. The 6-DOF sensor module comes with an I2C interface with a data rate of 400KHz. A serial peripheral interface for fast mode with a rate up to 20MHz is provided in the module. The MPU6050 find its applications in most of the intelligent electronics consumer products in one way or other. Virtual Reality gaming is the perfect application of MPU6050 6-DOF MEMS sensor module.

The MEMS sensor module consists of MPU6050 Chip, I/O Headers, pull-up resistors, LDO and Decoupling capacitors. The IC is integrated with an accelerometer and gyroscope does all the processing. The I/O Headers will serve the purpose of connectivity and data transmission with a micro controller unit. The pull-up resistors are placed for I2C bus to source current and establish a default state. LDO voltage regulator is provided to reduce the voltage to 3.3V for the IC functioning and increase power efficiency. The Decoupling capacitors remove the unwanted noise from the signals. The MPU6050 6-DOF MEMS sensor module has a total of 8 pins that includes INT pin, ECL pin, EDA pin, ADO pin, SCL pin, SDA pin, VCC and Ground.

Gyroscope Working Principle:

1. The Gyroscope which measures rotational velocity/rate of change of angular position over time works through the "Coriolis Effect" and MEMS technology.

   Coriolis Effect:

   The Coriolis force is an inertial force that acts on object in motion with in a frame of reference with respect to an inertial frame. In a reference frame with clock wise rotation, the force act on the left of the motion of the object and with anti-clock wise rotation, the force act on the right. The deflection of an object due to Coriolis force is known as "Coriolis Effect".

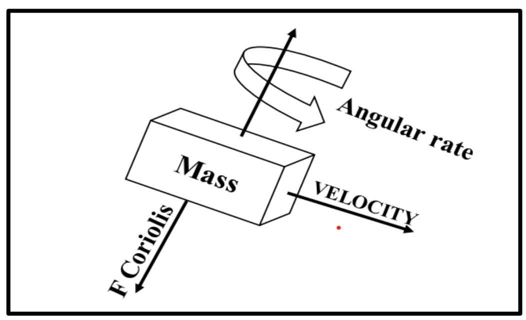


Fig 3.3 Coriolis Effect

2. When a mass is moving in a particular direction with a particular velocity and when an external angular rate will be applied as shown with the curved arrow a force will occur which the sensor will detect.
3. As shown in the figure, labelled by the arrows, this will cause perpendicular displacement of the mass. The displacement will cause a change in capacitance which will be measured, processed and it will correspond to a particular angular rate.
4. The outputs of the gyroscope are in degrees per second, so in order to get the angular position, we just need to integrate the angular velocity.

3.2 Interfacing

Interfacing Raspberry Pi 3 with MPU6050 Module

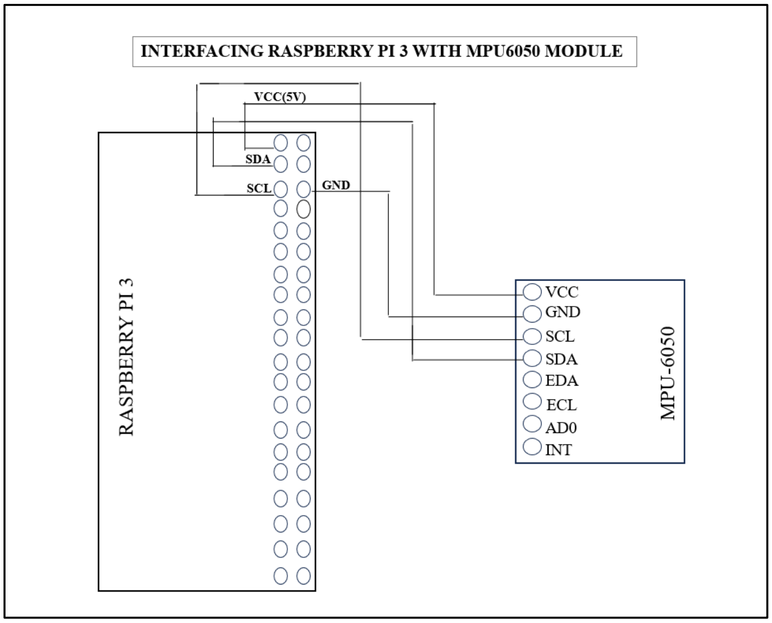


Fig 3.4 Interfacing

To connect a Raspberry Pi and an MPU-6050 gyroscope sensor, you need to make the appropriate physical connections and then enable the necessary software support.

Follow these steps:

1. Power Connections:
    1. Connect the VCC (power) pin of the MPU-6050 to the 5V pin on the Raspberry Pi.
    2. Connect the GND (ground) pin of the MPU-6050 to any ground pin on the Raspberry Pi.
2. I2C Connections:
    1. Connect the SDA (data) pin of the MPU-6050 to the SDA pin on the Raspberry Pi (GPIO2).
    2. Connect the SCL (clock) pin of the MPU-6050 to the SCL pin on the Raspberry Pi (GPIO3).

3.3 Software Implementation.

I. Install Raspbian OS on Raspberry Pi 3 Board.

Steps:

1. To install Raspbian OS on a Raspberry Pi, first install Raspbian OS on SD Card.
2. Open a browser tab, search for an SD Card formatter, click on the first link, and download the formatter suitable for your operating system (Windows/Mac).
3. In another browser tab, search for "etcher.io," and download the Etcher software for Windows or Mac.
4. Open a separate tab, visit the official Raspberry Pi website, navigate to the "Download" section, and select Raspbian. Download the ZIP file for Raspbian with Desktop.
5. Install the SD Card formatter and the Etcher software on your computer. Extract the contents of the downloaded Raspbian ZIP file.
6. Open the SD Card formatter, select your SD card, and format it.
7. Launch Etcher, choose the extracted Raspbian file, ensuring you select the correct SD card, and initiate the flashing process.
8. Upon completion of the flashing process, safely remove the SD card from your computer.
9. Insert the SD card into the Raspberry Pi board to initiate the Raspbian OS installation process.

II. Python Script for Knee Angle Measurement (Programming Language: Python)

1. Write a Python Script:
   1. Create a Python script that reads data from the gyroscope, processes it on raspberry pi and calculate the knee angle, and stores it in a data structure (e.g., a list).
2. Save Data to CSV (Comma-separated Values):
   1. Periodically save knee angle data to a CSV file. Use the ‘CSV’ module for this purpose.
   2. CSV is a simple file format used to store tabular data, where each row of the file represents a record, and the values within a row are separated by commas.

III. Excel Automation

1. Install pandas Library:

   Install the 'Pandas' library to facilitate working with data frames and Excel files.

2. Modify Python Script to Save to Excel:

   Adjust your Python script to convert the knee angle data to a pandas Data Frame and save it to an Excel file.

IV. Email Automation

1. Install smtplib Library:

   Install 'smtplib' library for sending emails.

2. Configure Email Settings:

   Set up an email account to be used for sending emails. Obtain the SMTP server details, username, and password.

3. Modify Python Script to Send Email

   Modify your Python script to send an email with the Excel file attached.

V. Classification

Using Random Forest Algorithm can categorize the person's condition as either normal or abnormal

3.3.1 Code:

1. Reading Knee angle data from the MPU6050 Gyroscope

```
from mpu6050 import mpu6050  # pip install mpu6050
import smtplib
import getpass
import pandas as pd
import time
import csv
from email.mime.text import MIMEText
from email.mime.multipart import MIMEMultipart
from email.mime.application import MIMEApplication

# MPU6050 I2C address
MPU6050_ADDRESS = 0x68

# Create MPU6050 object
mpu = mpu6050(address=MPU6050_ADDRESS)
```

```
# Calibrate the gyroscope
mpu.calibrate_gyroscope()

# Define the sampling rate and time interval
sampling_rate = 20  # Hz
time_interval = 1 / sampling_rate

# Initialize variables
previous_time = time.time()
angle = 0.0

# Open a CSV file for writing
csv_file = open('gyro_data.csv', 'w', newline='')
csv_writer = csv.writer(csv_file)
csv_writer.writerow(['Time (s)', 'Angular Velocity (X) (deg/s)', 'Angle (X) (degrees)'])

def create_excel_file(data, file_name='data.xlsx'):
    # Create a Pandas DataFrame
    df = pd.DataFrame(data)

    # Save the DataFrame to an Excel file
    df.to_excel(file_name, index=False)

def send_email(to_email, subject, body, attachment_path=None):
    # Email configuration
    sender_email = 'vidyasagar16k@outlook.com'  # your email
    password = getpass.getpass("Enter your email password: ")
    smtp_server = 'smtp-mail.outlook.com'  # your email provider's SMTP server
    smtp_port = 587

    # Create the email message
    message = MIMEMultipart()
    message['From'] = sender_email
    message['To'] = to_email
    message['Subject'] = subject

    # Attach body text
    message.attach(MIMEText(body, 'plain'))
```

```
    # Attach Excel file if provided
    if attachment_path:
        with open(attachment_path, "rb") as attachment:
            part = MIMEApplication(attachment.read(), Name="data.xlsx")
            part['Content-Disposition'] = f'attachment; filename={attachment_path}'
        message.attach(part)

    # Connect to the SMTP server and send the email
    with smtplib.SMTP(smtp_server, smtp_port) as server:
        server.starttls()
        server.login(sender_email, password)
        server.sendmail(sender_email, to_email, message.as_string())

try:
    while True:
        # Read angular velocity
        gyro_data = mpu.get_all_data()
        gyro_x = gyro_data[0].get('gyro', [0])[0]  # Assuming gyro[0] corresponds to the x-axis gyro data
        gyro_x = gyro_data[0].get('x', 0)

        # Calculate angle using integration
        current_time = time.time()
        elapsed_time = current_time - previous_time
        angle += gyro_x * elapsed_time
```

```
        # Print and write results to CSV
        print("Angular Velocity (X): {} deg/s".format(gyro_x))
        print("Angle (X): {} degrees".format(angle))
        csv_writer.writerow([current_time, gyro_x, angle])

        # Update previous time
        previous_time = current_time

        # Wait for the next sample
        time.sleep(time_interval)

except KeyboardInterrupt:
    print("Measurement stopped by user")
finally:
    # Close the CSV file
    csv_file.close()
    print("CSV file closed.")

    # Create Excel file from CSV data
    df = pd.read_csv('gyro_data.csv')
    df.to_excel('gyro_data.xlsx', index=False)

    # Email configuration
    to_email = input("Enter recipient email address: ")
    subject = 'Gyroscope Data Report'
    body = 'Attached is the gyroscope data in Excel format.'

    # Send email with the Excel attachment
    send_email(to_email, subject, body, 'gyro_data.xlsx')
    print("Email sent. Exiting program.")
```

2. Categorization of person's condition as either normal or abnormal.

```
import pandas as pd
import numpy as np
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

# Function to generate random data for a person
def generate_random_data():
    time = np.random.rand(5)
    angular_velocity = np.random.randint(0, 21, size=5)
    angles_of_movement = np.random.randint(0, 141, size=5)
    return {'time': time, 'angular_velocity': angular_velocity, 'angles_of_movement': angles_of_movement}

# Set the minimum and maximum angles
min_angle = 0
max_angle = 140

# Create DataFrames for 100 persons
persons_data = [generate_random_data() for _ in range(100)]
dfs = [pd.DataFrame(person_data) for person_data in persons_data]

# Calculate average angles for each person
avg_angles = [(df['angles_of_movement'].max() + df['angles_of_movement'].min()) / 2 for df in dfs]

# Add labels for each person (1 for normal, 0 for abnormal based on the average angle)
labels = [1 if avg_angle >= 60 else 0 for avg_angle in avg_angles]

# Concatenate DataFrames
df_combined = pd.concat([df.assign(label=label) for df, label in zip(dfs, labels)], ignore_index=True)

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(df_combined[['time', 'angular_velocity', 'angles_of_movement']], df_combined['label'], test_size=0.2, random_state=12)

# Create a Random Forest classifier
rf_classifier = RandomForestClassifier(n_estimators=100, random_state=42)

# Train the Random Forest model
rf_classifier.fit(X_train, y_train)

# Make predictions on the test set
y_pred = rf_classifier.predict(X_test)
```

```python
# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy: {accuracy}")

# Output for each person
for i in range(100):
    print(f"Data for Person {i + 1}:")
    print(dfs[i])

    output = 'Normal' if avg_angles[i] >= 60 else 'Abnormal'
    print(f"Condition: {output}\n")
```

3.4 Block Diagram

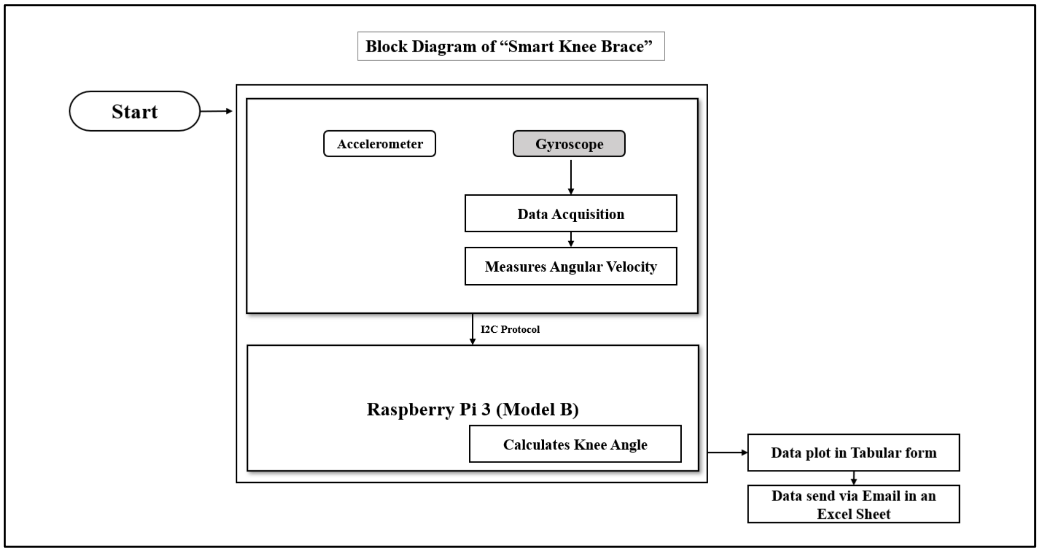


Fig 3.5 Block Diagram

Positioning of the sensor is crucial for obtaining an output suitable for the application in which the knee angle is to be measured. The location of the sensor needs to be such that it can pick up even minute movements. In order to quantify the knee swing movement in the sagittal plane, the sensor was positioned at that spot. The reference angular position of the knee joint served as the basis for the arrangement of the knee swing action.

There are several steps in the system's operation. The system operating procedure, which includes data acquisition, angular velocity measurement, Knee angle computation, tabular data visualization, and email-based Excel sheet data sharing, is depicted in the flowchart.

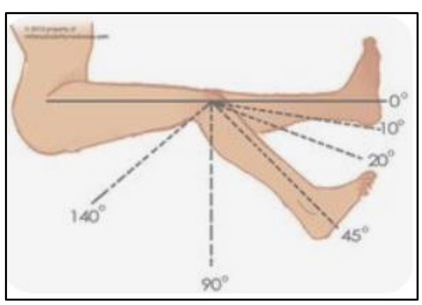


Fig 3.6 Arrangement of Knee Movements

The process of acquiring data initiated the system's functionality. The orientation of the MPU6050 Gyroscope sensor, which served as the input device, is crucial to the data indicating the variation in knee joint movement parameters. To get the necessary parameters, the data is then modified based on the reference sensor location. The parameters and intended final data output in this instance are the angular position and angular velocity of the knee joint movement.

I. Data Acquisition and Measurement of Angular Velocity

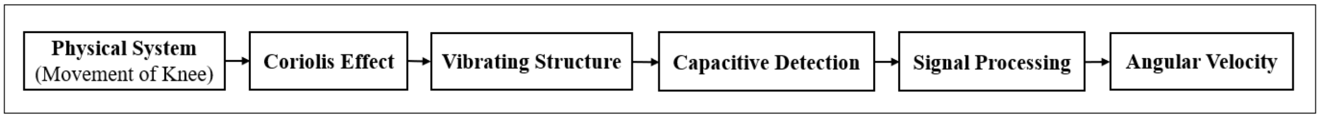


Fig 3.7 Flow Chart

The MPU6050 module is essential for measuring and identifying knee joint motion related to the "Coriolis Effect." This phenomenon occurs when a force perpendicular to the axis of rotation and the direction of motion acts on the gyroscope, which is a component of the knee's spinning mechanism. A vibrating structure, such a tuning fork or proof mass, experiences a deflection as an object fitted with a gyroscope moves within this rotating system.

A capacitive detecting system is used to harness the deflection of the vibrating structure for sensing purposes. There are variations in capacitance between the sensor's various components when the structure moves. The gyroscope-induced motion is directly shown by this variation in capacitance.

The first stage of the signal processing chain is started by converting the measured capacitance changes into an electrical signal. An essential part of the system, the MPU6050 module, is in charge of converting this electrical signal via an analog-to-digital conversion process into angular velocity. The MPU6050 uses complex algorithms and calibration factors to do this conversion precisely. These components are necessary to compensate for flaws and features unique to the sensor, guaranteeing the accuracy and dependability of the angular velocity data. The electrical signal that has been calibrated to indicate angular velocity.

The formula to calculate angular velocity (ω) from the sensor output is given by:

ω = (Sensor output – Zero Rate output) / Sensitivity

Where,

1. Sensor output means raw output from the gyroscope sensor
1. Zero-Rate output means output when there is no angular velocity (sensor at rest)
2. Sensitivity means the sensitivity of the gyroscope typically specified in units of angular velocity per unit of sensor output.

A Gyroscope in the MPU6050 is indeed a 3- axis device, meaning it can measure angular velocity along three axes (usually labelled as X, Y, Z).

Mathematically representation of Angular Velocity (ω) is

$$\boldsymbol{\omega} = \frac{\boldsymbol{d\theta}}{\boldsymbol{dt}}$$

Where,

1. ω is Angular Velocity
2. θ is Angular displacement
3. t is time

In our gyroscope-based knee joint motion detection system, the sampling rate—measured in hertz (Hz)—is a crucial parameter that establishes the frequency at which the gyroscope takes measurements throughout time. Essentially, it represents the quantity of samples obtained every second, providing information on the level of detail in the data collection. Selecting a 2KHz sampling rate means that the gyroscope may capture a significant 2000 samples per second for each axis, which enables a very accurate depiction of the dynamic motion of the knee joint. But it's crucial to remember that although a faster sampling rate increases data precision by providing more data points, it can also increase power consumption and processing needs. For this reason, we had to carefully balance accuracy and resource usage when designing our system.

Sampling time or period, represented by T, is a term that is complementary to sampling rate. This temporal metric, which shows the interval between successive measurements, is the inverse of the sampling rate. Mathematically, T = 1/f gives the sampling time if the sampling rate is represented as f in hertz. In our particular instance, the relevant sampling time is determined as T = 1/2000, yielding an astonishingly brief length of 0.0005 seconds, with a selected sampling rate of 2000Hz. This suggests that our method, which takes successive measurements every half a millisecond to ensure a high temporal resolution in our data collecting process, rapidly captures the fine details of knee joint motion.

II. Knee angle Calculation

By interfacing Raspberry pi with MPU6050 module, Raspberry pi can read data (Angular Velocity) from the gyroscope using I2C protocol (Inter- Integrated Circuit). Libraries such as “smbus” can be used for I2C communication.

Formula to calculate angle from angular velocity by Raspberry pi

Angle += Gyro _x * time interval

Where,

1. Angle is the variable represents the accumulated or integrated angle. It starts with an initial value (presumably 0) and gets updated over time.
2. Gyro _ x is the variable represents the angular velocity around the X-axis. The gyroscope sensor provides this value in degrees per second (deg/s) or radians per second (rad/s), depending on the sensor configuration.
3. Time interval is the variable represents the time elapsed since the last iteration of the loop. It's calculated as the difference between the current time and the previous time.

Calculating knee angle from angular velocity data collected from a gyroscope involves integrating the angular velocity over time. The angular displacement (θ) can be obtained by integrating the angular velocity (ω) with respect to time (t).

This is mathematically represented as:

Numerical Integration:

$$\boldsymbol{\theta(t) = \int_{t_0}^{t} \omega(t)\,dt}$$

III. Data plot in tabular form (CSV Format)

After calculating angles using angular velocity data from a gyroscope, display the outcomes in a tabular format.

Time(sec)	Angular Velocity(deg/sec)	Angle(deg)
0.0	5.2	0.0
0.1	6.8	0.52
0.2	3.5	1.19

Table.3.1 Data plotted in tabular format

i. **Time (seconds):** The timestamp or time interval at which the data is recorded.

ii. **Angular Velocity (°/s):** The measured angular velocity at each time point.

iii. **Calculated Angle (degrees):** The angle obtained by integrating angular velocity data up to each time point.

IV. Data sent to email in an excel sheet

Write a python script that utilizes the smtplib library to send an email containing data in an excel sheet. The excel sheet, created using the panda’s library, represents time, angular velocity, and calculated angles.

3.5 Classification

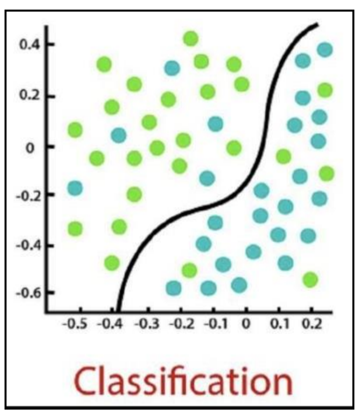


Fig 3.8 Classification

An essential part of machine learning is a classification algorithm, which is used to label or classify input data into pre-established groups or categories. Finding patterns in the training data to identify a mapping from input features to different output classes is the main goal. Numerous fields, including as image identification, natural language processing, and medical diagnosis, use classification methods. In order to identify patternsand relationships in the data, these algorithms make use of mathematical models, statisticalmethods, or neural networks. This allows them to generalize and precisely forecast the class labels of new, unseen cases. Depending on the nature of the problem, the properties of the data, and the intended balance between interpretability and model complexity, a particular classification technique may be used. Classification algorithms include decision trees, logistic regression, support vector machines, and ensemble techniques like gradient boosting and **Random Forests**. Metrics like recall, accuracy, precision, and F1 score are frequently used to evaluate the efficacy of classification algorithms and provide information about how well they perform and fit for a certain task.

Categorize individual's condition as either normal or abnormal

Random Forest Algorithm

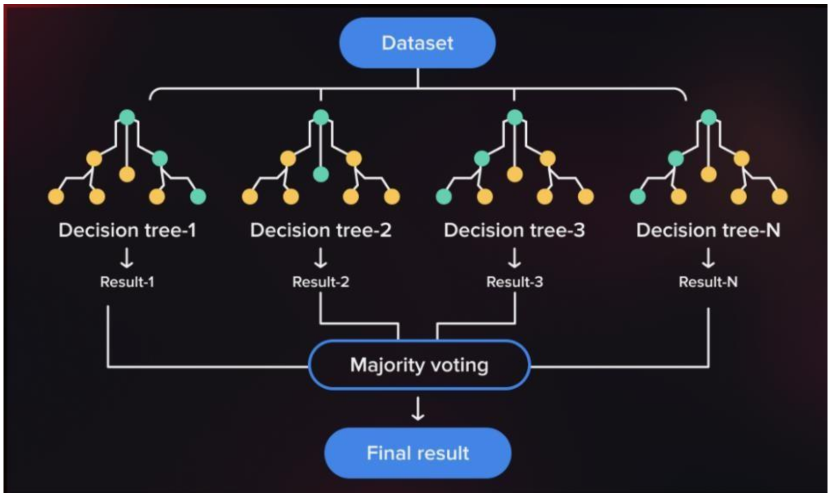


Fig 3.9 Random Forest Algorithm

Using characteristics like the minimum and maximum angles of movement, a classification modelfor determining a person's state (Normal or Abnormal) is constructed using the Random Forest method.

To begin, the Random Forest is an ensemble learning method that builds several decision trees while it is being trained. A random selection of the dataset and a random selection of features areused to train each decision tree. This unpredictability makes the model more resilient and less prone to overfitting by decorrelating the individual trees.

In the testing stage, the Random Forest also uses a majority voting process to aggregate the predictions made by each individual tree. When compared to a single decision tree, this ensemblemethod frequently produces better generalization results. Enhancing the model's capacity to manage a range of Normal and Abnormal circumstances, it aids in capturing the numerous patternsfound in the dataset.

Furthermore, the Random Forest inherently provides a feature importance measure, allowing practitioners to understand which features (angles, in this case) contribute more significantly to the classification. This information can be valuable for interpreting the model and gaining insightsinto the relevant aspects of the data.

Finally, by utilizing its ensemble nature, resilience, and feature significance capabilities, the Random Forest algorithm is effective in this code. It has the ability to produce precise and broadly applicable forecasts and offers a dependable framework for utilizing movement angles to categorize an individual's state.

Steps

1. Data Collection

Get movement data from an individual by monitoring their lowest and maximum angles of movement while performing particular tasks, such as sitting. The dataset's recorded instances are then classified as "Normal" or "Abnormal" according to a predetermined threshold, whichis frequently set at 60 degrees. This threshold is used as a criterion to categorize motions as either abnormal or falling outside of the predicted range. For example, motions with average angles less than 60 degrees may be seen as normal, whereas movements with average angles more than 60 degrees may be regarded as abnormal. In order to train machine learning models, like Random Forests, to automatically identify and forecast the condition of a person'smovements based on the supplied angle features, labeling is necessary. This procedure yields a labeled dataset.

Data Set

Minimum angle	Maximum angle	Condition
45	65	Abnormal
55	75	Normal
40	60	Abnormal
50	72	Normal
60	80	Normal
43	63	Abnormal
52	72	Normal
58	68	Abnormal
56	76	Normal

Table 3.2 Data Set

2. Data Preprocessing

The input data is put through necessary changes during the preprocessing phase to make sure it is suitable for training a machine learning model. Feature scaling and handling missing variables are two essential preprocessing methods.

1. Handling Missing Values

Missing values are frequently found in real-world datasets, which can negatively affect how well machine learning models perform. Preprocessing techniques like imputation, in which missing data are filled in with estimated or calculated values, are used to remedy this. For example, methods like mean imputation or utilizing neighboring data points maybe used to fill in the gaps if the dataset lacks entries for the minimum or maximum angles.Managing missing values well guarantees a complete and trustworthy dataset for further model training.

2. Scaling Features

There may be large variations in the dataset's range of values for certain attributes. Featurescaling is a preprocessing technique that ensures no feature dominates the learning process because of its higher magnitude by normalizing the size of features. Standardization (Z- score normalization) or Min-Max scaling are popular techniques for feature scaling. Scaling prevents bias toward features with greater numeric ranges by ensuring that the minimum and maximum angle values are displayed proportionately in the context of movement angle data. This is an important step, particularly when using algorithms that rely on distance or that are sensitive to the size of the input characteristics.

3. **Train Random Forest Classifier**

The dataset is divided into the training set and the out-of-bag set during the Random Forest classifier's training phase. In particular, the Random Forest method, which is made up of several decision trees, cannot be created without the training data. By using a technique called bootstrap sampling, each decision tree in the forest is trained separately using a random subset of the training set. Using bootstrap sampling, instances with replacement are chosen at randomfrom the training set. This means that some data points may be included more than once whileothers may be excluded. As a result, every decision tree becomes distinct and picks up new information about various facets of the total data distribution. One of the Random Forest's mainadvantages is its ability to capture a wider variety of patterns and relationships in the data thanks to the diversity of the trees. Because of this, the group of decision trees works togetherto create a strong, broad model that can accurately predict outcomes on fresh, untested data.

4. **Decision Trees**

When a Random Forest is made up of individual decision trees, it functions by assembling a group of different learners, called decision trees. During the training phase, Decision Tree 1, Decision Tree 2, and subsequent trees are individually generated in the context of classifying an individual's condition based on movement angles. The minimum and maximum angles of movement are two of the variables in the dataset that these decision trees take into account while making judgments. The decision trees assess these features for each occurrence in the dataset and then classify the cases as "Normal" or "Abnormal" based on a set of branching conditions. Because of the randomization included during training, every tree offers a new viewpoint and recognizes various patterns in the data. The Random Forest enhances its capacity to deliver correct and dependable classifications for a variety of input instances by utilizing the collective intelligence of its constituent decision trees, which are aggregated by majority voting. This method keeps the model from overfitting by limiting its dependence on the subtleties of individual trees, while still guaranteeing robustness.

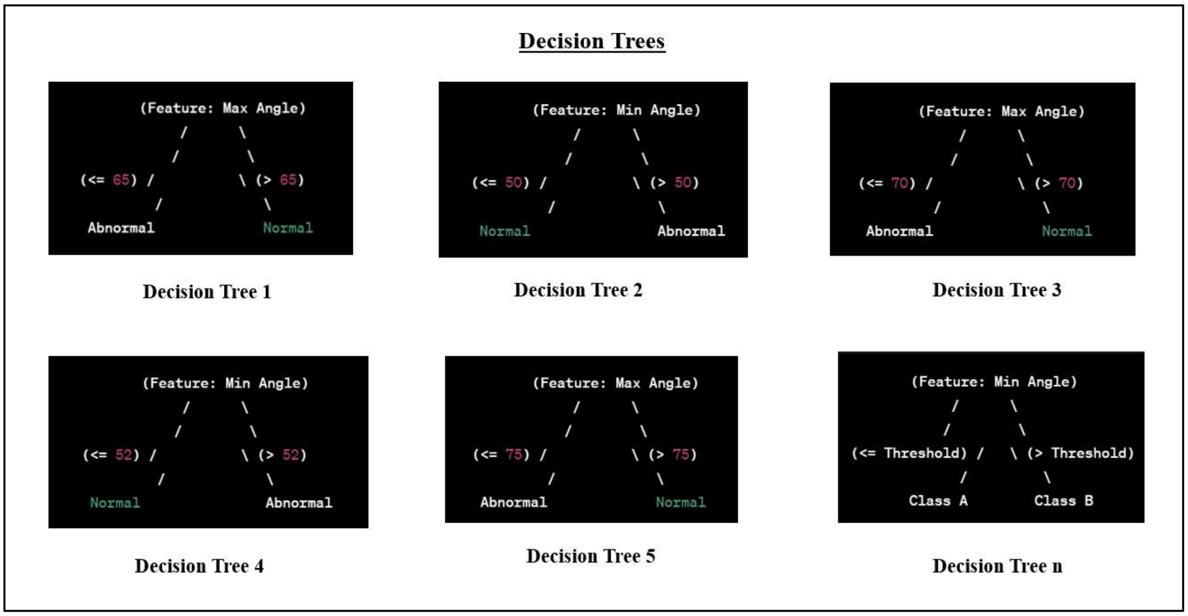


Fig 3.10 Decision Trees

5. Combination of Results

For a new instance, each tree makes a prediction. Let's consider a new instance with minimumangle = 53 and maximum angle = 76.

Tree 1: Predicts Normal

Tree 2: Predicts Abnormal

Tree 3: Predicts Normal

Tree 4: Predicts Abnormal

Tree 5: Predicts Normal

Majority Vote: Normal

6. Classification Threshold Check

Calculate the average angle for the new instance:

Average Angle = (Min Angle + Max Angle)/2

Average Angle = (53+76)/2 = 64.5

Since 64.5 ≥ 60, we classify the instance as Normal.

7. Output

The final output is Normal based on the majority vote and the classification threshold.

4. RESULT AND DISCUSSION

4.1 Experimental Setup

Performance of the developed “Smart Knee Brace” was evaluated on knee swing movement, where a subject was set to be in sitting position. Measuring an individual's range of motion entails determining their minimum and maximum movement angles. A predefined range is defined for an average person, with a minimum angle of movement of 0° and a maximum angle of movement capped at 140°. These particular cutoff points function as standards for grouping people into discrete conditions according to their tracked movements. The average angle between the minimum and maximum angles is computed to provide the overall characterization. When the average angle is greater than 60°, the subject's movement is considered to be within the usual range and is considered to be in a normal condition. On the other hand, if the average angle is less than 60°, it may indicate that the person moves abnormally. With the use of the average angle as a critical signal, this methodical technique facilitates a clear and quantitative assessment, allowing individuals to be classified as normal or abnormal based on the observed movement angles.

4.2 Results

4.2.1 Data Received via Email

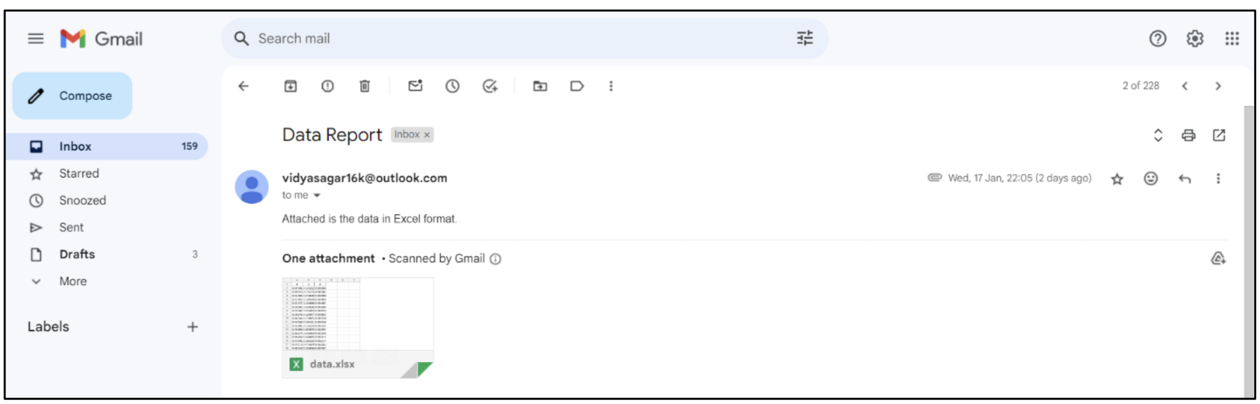


Fig 4.1 Data Received via Email

4.2.2 Knee Angle Movement Data

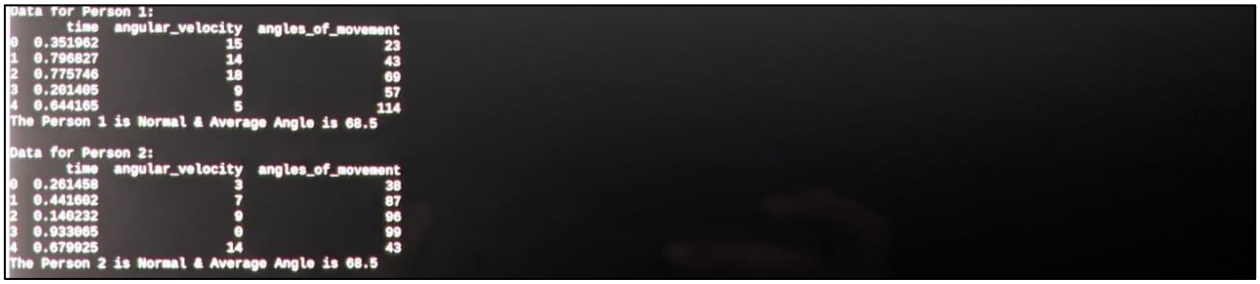
1	Time (sec)	Angular Velocity (deg/sec)	Angle (deg)
2	0.00456738472	0.7295343511	0.0003647671756
3	0.00456738472	12.34022137	0.006534877863
4	0.00456738472	20.35548855	0.01671262214
5	0.00456738472	21.56922901	0.02749723664
6	0.00456738472	20.30968702	0.03765208015
7	0.00456738472	16.40129008	0.04585272519
8	0.00456738472	9.538694656	0.05062207252
9	0.00456738472	2.157015267	0.05170058015
10	0.00456738472	1.178862595	0.05111114885
11	0.00456738472	2.056725191	0.05008278626
12	0.00456738472	0.06435877863	0.05005060687
13	0.05604815483	0.5005267176	0.05030087023
14	0.06121182442	0.6150305344	0.0506083855
15	0.06610870361	5.522374046	0.04784719847
16	0.07113909721	12.8658855	0.04141425573
17	0.07596611977	22.2399313	0.03029429008
18	0.08084583282	31.89641985	0.01434608015
19	0.08562755585	38.79718321	0.00505251145
20	0.09100699425	44.301	0.02720301145
21	0.09504628181	31.58344275	0.04299473282
22	0.09916114807	9.881152672	0.04793530916
23	0.1033601761	2.598709924	0.04923466412
24	0.1079831123	0.5005267176	0.04898440076
25	0.1133098602	0.9651221374	0.04946696183
26	0.1184720993	7.988022901	0.05346097328

Table 4.1 Knee angle data of person 1.

1	Time (sec)	gular Velocity (deg/s	Angle (deg)
2	0.004543781281	0.1612270992	0.00008061354962
3	0.009206533432	0.275730916	0.0002184790076
4	0.01386237144	0.2299293893	0.0003334437023
5	0.01895546913	0.1917614504	0.0004293244275
6	0.02364087105	0.2222958015	0.0005404723282
7	0.02830982208	0.1535935115	0.000617269084
8	0.03302359581	0.2222958015	0.0007284169847
9	0.03819227219	0.1001583969	0.0007784961832
10	0.04282927513	0.0009217557252	0.0007789570611
11	0.04737591743	0.3367996183	0.0009473568702
12	0.05202937126	0.1001583969	0.0009974360687
13	0.05674290657	0.1841278626	0.0010895
14	0.06196951866	0.1764942748	0.001177747137
15	0.06690001488	0.4055019084	0.001380498092
16	0.07177400589	0.3444332061	0.001552714695
17	0.07650923729	0.3444332061	0.001724931298
18	0.08127093315	0.2222958015	0.001836079198
19	0.0860657692	0.1535935115	0.001912875954
20	0.09079313278	0.1612270992	0.001993489504
21	0.09548592567	0.02382251908	0.002005400763
22	0.1001946926	0.2451965649	0.002127999046
23	0.1037652493	0.1993950382	0.002227696565
24	0.1070623398	0.1383263359	0.002296859733
25	0.110640049	0.1993950382	0.002396557252
26	0.1141271591	0.1001583969	0.00244663645

Table 4.2 Knee angle data of person 2

5.2.2 Categorization



```
Data for Person 1:
       time  angular_velocity  angles_of_movement
0  0.351962                15                  23
1  0.796827                14                  43
2  0.775746                18                  69
3  0.201405                 9                  57
4  0.644165                 5                 114
The Person 1 is Normal & Average Angle is 68.5

Data for Person 2:
       time  angular_velocity  angles_of_movement
0  0.261458                 3                  38
1  0.441602                 7                  87
2  0.140232                 9                  96
3  0.933065                 0                  99
4  0.679925                14                  43
The Person 2 is Normal & Average Angle is 68.5
```

Fig 4.2 Categorization of person's condition

The output provides a performance and the data for each person:

1. "Accuracy: 0.95" indicates the accuracy of the Random Forest classifier in predicting the conditions.

2. Each person's data is displayed with columns for time, angular velocity, and angle of movement.

3. The data is provided for each of the 2 persons generated in the code.

4. The condition for each person is labeled as either "Normal" or "Abnormal" based on their average angle.

5. The condition is determined by whether the average angle is greater than or equal to 60 degrees.

6. The accuracy score represents the model's overall performance in classifying the conditions.

7. The output includes a mix of persons labeled as "Normal" and "Abnormal" based on their average angles.

8. Overall, the output provides insights into how the model classifies the conditions for each person based on the generated data.

5. CONCLUSION AND FUTURE SCOPE

This study presents the design and implementation of a "Smart Knee Brace," utilizing a single MPU6050 gyroscope interfaced with a Raspberry Pi 3 to measure knee angles with respect to time. The knee joint's intricate movements during extension and flexion lead to a continuous knee swing. The devised system adeptly captured angular velocity concerning the time frame, following a predefined algorithm within the Raspberry Pi 3. The resultant data, displayed in tabular form, was efficiently transmitted via designated email in an excel sheet, showcasing variations across diverse movement setups. Disparities in knee joint torque, applied during the movement, potentially influenced the recorded angular velocities. Further exploration is envisioned to assess the "Smart Knee Brace's" capacity to measure sound during knee joint movement and extend its applicability to diverse activities like sitting to stance, walking, or using exercise equipment such as cycling and elliptical stepping. Future endeavors include implementing artificial algorithms to optimize the MPU6050 gyroscope's singular utility in applications that traditionally necessitate the integration of two gyroscope sensors.

Future Scope

The project's future goals include improving the ability to evaluate the gyroscope data's movement patterns and ascertain the user's physical condition. In particular, the goal is to use the minimum and maximum angles found throughout the data gathering procedure to compute the average angle of movement. We'll use the Random Forest method to do this. The gyroscope data will be analyzed by this algorithm, which is renowned for its adaptability and capacity to manage intricate relationships within data, in order to produce insightful results.

6. REFERENCES

[1] R. V. Vitali et al., “Method for estimating three-dimensional knee rotations using two inertial measurement units: Validation with a coordinate measurement machine,” Sensors (Switzerland), vol. 17, no. 9, 2017, doi: 10.3390/s17091970.

[2] Y. P. Huang, Y. Y. Liu, W. H. Hsu, L. J. Lai, and M. S. Lee, “Monitoring and assessment of rehabilitation progress on range of motion after total knee replacement by sensor-based system,” Sensors (Switzerland), vol. 20, no. 6, 2020, doi: 10.3390/s20061703.

[3] J. Z. Edwards, K. A. Greene, R. S. Davis, M. W. Kovacik, D. A. Noe, and M. J. Askew, “Measuring flexion in knee arthroplasty patients,” J. Arthroplasty, vol. 19, no. 3, pp. 369–372, 2004, doi: 10.1016/j.arth.2003.12.001.

[4] D. Bennett, B. Hanratty, N. Thompson, and D. Beverland, “Measurement of knee joint motion using digital imaging,” Int. Orthop., vol. 33, no. 6, pp. 1627–1631, 2009, doi: 10.1007/s00264-008-0694-9.

[5] K. H. Somruthai Poomsalood1, 2, Karthik Muthumayandi1, “Can stretch sensors measure knee range of motion in healthy adults?,” p. 8, 2019, doi: 10.2478/bhk-2019-0001.

[6] G. M. Salim and M. A. Zawawi, “Knee joint movement monitoring device based on optical fiber bending sensor,” J. Telecommun. Electron. Comput. Eng., vol. 10, no. 1–3, pp. 25– 29, 2018.

[7] C. J. Chen, Y. T. Lin, C. C. Lin, Y. C. Chen, Y. J. Lee, and C. Y. Wang, “Rehabilitation System for Limbs using IMUs,” Proc. - Int. Symp. Qual. Electron. Des. ISQED, vol. 2020-March, no. March, pp. 285–291, 2020, doi: 10.1109/ISQED48828.2020.9137026.

[8] S. Cordillet, N. Bideau, B. Bideau, and G. Nicolas, “Estimation of 3D knee joint angles during cycling using inertial sensors: Accuracy of a novel sensor-to-segment calibration procedure based on pedaling motion,” Sensors (Switzerland), vol. 19, no. 11, 2019, doi: 10.3390/s19112474.

[9] V. Bonnet, V. Joukov, D. Kulić, P. Fraisse, N. Ramdani, and G. Venture, “Monitoring of Hip and Knee Joint Angles Using a Single Inertial Measurement Unit during Lower Limb Rehabilitation,” IEEE Sens. J., vol. 16, no. 6, pp. 1557–1564, 2016, doi: 10.1109/JSEN.2015.2503765.

[10] S. Pereira and M. De Oliveira, “KneeRecovery Rehabilitation Exercises for Knee Recovery at Home,” 2017, [Online]. Available: https://repositorioaberto.up.pt/bitstream/10216/105101/2/198424.pdf.

[11] T. Gujarathi and K. Bhole, “Gait Analysis Using Imu Sensor,” 2019 10th Int. Conf. Comput. Commun. Netw. Technol. ICCCNT 2019, no. July 2019, 2019, doi: 10.1109/ICCCNT45670.2019.8944545.

[12] R. Antunes, P. Jacob, A. Meyer, M. A. Conditt, M. W. Roche, and M. A. Verstraete, “Accuracy of measuring knee flexion after tka through wearable imu sensors,” J. Funct. Morphol. Kinesiol., vol. 6, no. 3, 2021, doi: 10.3390/jfmk6030060

[13] World Health Organization: https://en.wikipedia.org/wiki/World_Health_Organization

[14] National Institute of Health: https://en.wikipedia.org/wiki/National_Institutes_of_Health

[15] IEEE (It is the world's largest technical professional organization dedicated to advancing technology for the benefit of humanity): to know about Raspberry pi and MPU9250 Module

[16] Python for Signal Processing: José Unpingco

[17] Python for Data Analysis: Wes McKinney

[18] Python for computational Science and engineering: Hans Fangohr