Universiteit Leiden

Master Computer Science

An Algorithm for Identifying Activity Tests in Parkinson's Disease Patients using Accelerometer Data

Name:	Mouni Priyanka Thandu
Student ID:	s3420361
Date:	28/02/2025
Specialisation:	Data Science
1st supervisor:	Dr. Vasileios Exadaktylos
2nd supervisor:	Dr. Lu Cao

Master's Thesis in Computer Science

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Abstract

This research focuses on developing an algorithm for identifying structured activity tasks as part of PROPARK study in Parkinson's Disease (PD) patients using wrist-worn accelerometer data. Currently, researchers rely on manual annotation to find the start and end times of the tasks, which is time-consuming and prone to inter-observer variability. We developed a two-part algorithm with the Tap Detection and the Activity Task Marker modules to address this problem. The Tap Detection module detects the identifier taps performed by the patient before the tasks to indicate the beginning of the task and the Activity Task Marker module, which detects the start and end times of a task using envelope extraction methods. The algorithm employs task specific logic to use either standard deviation or variance envelopes and between accelerometer or gyroscope data based on task characteristics.

We validated the algorithm using two datasets: a patient-annotated diary dataset (251 participants) and an expert-annotated dataset(15 participants). The results show that our algorithm can reliably pick out where the task begins and ends without human intervention. This algorithm offers a scalable and consistent solution for long-term monitoring and studying the PD patients during the structured tests. This can be highly useful in tracking symptoms and evaluating treatment effects over time.

Acknowledgments

I would like to express my sincere appreciation to the following people and institutions for their invaluable assistance in finishing my thesis:

1. **Dr. Vasileios Exadaktylos:** I express my deepest gratitude to Dr. Vasileios Exadaktylos for all of his help and support. His guidance has been tremendously beneficial, offering vital perspectives that have profoundly influenced the course of this research.

2. **Dr. Lu Cao:** I would especially want to express my gratitude to Dr. Lu Cao for their unwavering support during this academic endeavor. Her support and knowledge have been a source of inspiration, giving the research more substance and authority.

3. **Dr. Roel Weijer:** I am appreciative to Dr. Roel Weijer for his constant guidance and support in assisting me in understanding the subject and navigating challenging data. His advice has been extremely helpful in improving the quality and relevance of this research.

4. **Leiden University Medical Center (LUMC) & Center for Human Drug Research (CHDR):** I express my gratitude to LUMC and CHDR for their invaluable resources and assistance during this investigation. The company's resources and data access have greatly increased the breadth of my research.

Their combined efforts have been important in ensuring that the research has been completed successfully, and I deeply appreciate their unwavering support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Parkinson's disease (PD) is a progressive neurodegenerative disorder that affects approximately 1% of the population over the age of 60 years Samii et al. (2004). Clinical signs and symptoms of Parkinson's disease (PD), such as bradykinesia, stiffness, and tremors, are indicative of the degeneration of dopaminergic neurons in the substantia nigra and are used to make the diagnosis [Bickley & Szilagyi (2012)]. To effectively manage the effects of Parkinson's disease (PD) on daily activities, constant monitoring is necessary. Although the clinical diagnosis of Parkinson's disease (PD) is fairly precise, objective and reliable indicators of the disease progression are necessary to track patients' responses to therapy and support the creation of novel treatments.

"PROfiling PARKinson's disease" (PROPARK) is a large-scale research initiative in the Netherlands that investigates the onset and progression of Parkinson's Disease, as well as the side effects of the medications. This study is a joint initiative between multiple Dutch medical institutions, including Leiden University Medical Center (LUMC), Amsterdam UMC (AMC and VUmc locations), Erasmus Medical Center, and Meander Medical Center in Amersfoort. It also involves partnerships with the Parkinson's Association, NL, Centre of Human Drug Research, NL, and both commercial and non-commercial organizations. The PROPARK aims to develop more personalized treatments by collecting and analyzing data from a large cohort of Parkinson's patients. Over 300 patients have already registered in PROPARK. The study is ongoing and has the potential to have a significant impact on the area of Parkinson's disease research. ProParkinson website provides valuable information on Parkinson's disease and the PROPARK study [1]. The study collects data from a large cohort of patients who perform a standardized test battery at home without supervision while wearing accelerometers [LeMoyne (2013)]. The standardized test battery here refers to the series of 8 tasks called motor tests that patients perform twice a day, once before and once after medication or an hour between assessments if the par-

[1] `https://proparkinson.nl`

ticipant does not take PD medication, to assess their symptoms. The data collected includes recordings from an accelerometer [Axivity Ltd. (n.d.)] at the wrist and one at the lower back and information on medication usage. This research aims to leverage the accelerometer data from the study and develop an algorithmic solution to identify key markers in PD patients' activity tests. Developing an automated algorithm will minimize the influence of inter-observer variability, reduce the manual effort required to annotate the data, and provide consistent and reliable monitoring of PD patients' activities. This algorithm will help enhance the accuracy of activity monitoring, and enable better disease management and data-driven personalized medicine.

1.2 Research Question and Objectives

Research Question

How can accelerometer data be effectively utilized to identify specific markers, such as pre-activity taps and task initiation/termination, in PD patients' activity tasks?

Objectives

The main objectives of this research are:

1. To develop an algorithm to detect the pre-activity taps performed by PD patients.
2. To identify markers indicating the initiation and termination of activity tasks using accelerometer data.
3. To evaluate the accuracy and efficiency of the proposed algorithm.

1.3 Structure of the Thesis

The rest of this thesis is organized as follows:

- **Chapter 2** provides a literature review.
- **Chapter 3** describes the methodology.
- **Chapter 4** presents the insights into the results.
- **Chapter 5** provides a conclusion of the study.

Chapter 2

Literature Review

2.1 Background on Parkinson's Disease

Parkinson's disease or PD, is a complicated neurological disorder that greatly affects an individual's mobility and quality of life [Kandel et al. (2000)]. Since the first mention of it in 1817 by the English physician James Parkinson, the illness has been the focus of extensive medical research. Fundamentally, Parkinson's disease can be described as the slow degeneration of dopamine-producing neurons in the substantia nigra, a region of the brain [Auluck & Bonini (2002)]. An essential neurotransmitter that controls movement and coordination is dopamine. The brain finds it more difficult to properly regulate and coordinate bodily movements when these cells deteriorate. Parkinson's disease presents a unique set of challenges due to the vast disparity in symptoms across people. Tremors, bradykinesia, stiffness, slowness of movement, and postural instability are typical symptoms [Bickley & Szilagyi (2012)]. In addition, the disease's effects on day-to-day functioning might be further complicated by non-motor symptoms like mood swings, sleep difficulties, and cognitive impairment. Parkinson's disease is progressive, meaning that symptoms worsen over time. While there is currently no cure, medical and therapeutic interventions aim to manage symptoms and enhance the quality of life for those affected. Every individual experiences the disease differently; some worsen rapidly compared to others, and some decline at a slower rate.

Parkinson's disease research is dynamic and ongoing. Researchers from all across the world are pursuing a variety of avenues, from comprehending the genetic components of the illness to developing innovative therapies like deep brain stimulation [Volkmann et al. (2006)]. Although a cure is still unattainable, advances in understanding the disease and technology provide hope for better outcomes and treatment. Monitoring the symptoms can help researchers track the disease progression and assess treatment. Wearable sensor technology, particularly Accelerometers helps monitor PD symptoms continuously and objectively. Using the high-resolution motion data obtained from the Accelerometers, the researchers can analyze movement patterns in real-world settings. However, processing the Accelerometer data to interpret

meaningful information from the sensors requires robust signal-processing techniques.

2.2 Existing Research

2.2.1 In Signal Processing

Signal processing plays a crucial role in extracting meaningful patterns from raw acceleromter data. Low-pass filtering is commonly used to eliminate high-frequency noise while maintaining the crucial low-frequency components associated with movement. They are designed to attenuate higher-frequency components while allowing low-frequency components to flow through. They are commonly used in audio processing, image processing, and communications to isolate and amplify the lower-frequency components. Over the years many tools have been employed to deal with high-frequency noise, detailed discussions on these filters are provided by Winter (1990). Butterworth filters, among all the available filters, are widely accepted because of their simplicity and satisfactory performance. The Butterworth filter is a type of infinite impulse response (IIR) filter. Robertson & Dowling (2003) discusses the design of the Butterworth filters. The cut-off threshold frequency is the key parameter that influences the design of the Butterworth filter. The user sets the cut-off threshold frequency based on their requirement. The order of a Butterworth filter is a measure of its complexity. It dictates how quickly the filter attenuates higher-frequency components beyond the cut-off frequency. The higher order is used in situations, like communication systems, where exact control over the frequency response and quick attenuation of higher frequencies are important. The Butterworth filter's transfer function uses a smooth response curve to reduce high-frequency noise without distorting the overall movement's shape. The Butterworth transfer function in the Laplace domain has the following general form:

$$H(s) = \frac{1}{1 + \left(\frac{s}{\omega_c}\right)^{2N}} \tag{2.1}$$

where $H(s)$ is the transfer function, s is the complex frequency variable, ω_c is the cutoff frequency and N is the filter order.

According to the book by Johnson Jr et al. (2011), the envelope of a signal is a useful feature in various signal processing applications. It can provide insights into the signal's amplitude variations over time. Envelope extraction can help us identify the changes in the movement intensity which can help with event-based segmentation. By computing the standard deviation or variance within a moving window, envelope extraction highlights the fluctuations in movement and helps in event detection. Standard deviation-based envelope extraction is renowned for its sensitivity to variations in signal amplitude. This approach has been used for signal analysis by several researchers over the years, including Nygårds & Sörnmo (1983); Silva et al.

(2009); Thalmayer et al. (2020). An equally sophisticated technique for extracting the signal envelope is to use variance within a moving window. Distinct from standard deviation, variance offers a complementary insight into signal spread and is robust to outliers. The researchers Weinstein & Weiss (1984); Cespedes et al. (1995); Varghese & Ophir (1998) highlight that the signal envelope may be extracted by employing the variance of the signal values.

2.2.2 In Parkinsons Research

In PD research, accelerometer data plays a crucial role in monitoring disease progression, assessing treatment efficacy, and aiding rehabilitation efforts. Signal processing techniques such as low-pass filtering and envelope extraction can be used on the accelerometer datasets for enhancing data quality and identifying relevant features. These techniques are used to preprocess the raw accelerometer data and help reduce the noise, improve the signal-to-noise ratio, and enable accurate detection of motor symptoms in PD patients. Tremor detection in PD has been widely studied using frequency-based methods. For example, Keijsers et al. (2006) developed an ambulatory motor assessment system to monitor PD symptoms using accelerometers. Their approach to tremor detection involved filtering accelerometer data using a band-pass filter tuned to the 4-6 Hz range, where Parkinson's tremors typically occur. They then applied peak detection to identify rhythmic oscillations in the filtered data, which corresponded to tremor events. This method allowed them to not only detect tremors but also quantify their severity by analyzing the amplitude of these oscillations.

Detecting peaks in accelerometer signals is another significant way for recognizing key events or changes in motion patterns, particularly in event-based analysis such as detecting tremors or frozen episodes in Parkinson's disease. By analyzing the magnitude of x, y, and z signals, researchers can detect peaks that correspond to specific activities or motor symptoms. In their study on throwing event detection using wrist-worn sensors, Schweiger et al. (2020) employed a method based on the magnitude of acceleration signals to identify peaks corresponding to throwing events. They calculated the magnitude of the accelerometer data using the following equation:

$$M = \sqrt{x^2 + y^2 + z^2} \tag{2.2}$$

The magnitude was then analyzed to detect peaks, which were indicative of throwing events. The authors applied a threshold-based method to differentiate between significant peaks and noise, allowing for accurate detection of throwing activities. This methodology underscores the utility of acceleration magnitude in capturing distinct motion patterns, which can be extended to other domains such as PD research for detecting tremors or gait abnormalities.

2.3 Gap Identification

While existing studies provide valuable insights into PD monitoring, a clear gap exists when it comes to automatically detecting structured task-based activities like those in PROPARK study. Most of the previous studies focus on unstructured continuous movement analysis rather than the detection of predefined motor tasks. The existing peak detection and frequency analysis approaches seem to work for tremor detection but they do not explicitly segment task start and end times for the motor tasks.

The PROPARK researchers currently rely on manual annotations for labeling the motor tasks, which can be time-consuming and prone to inter-observer variability. The patients perform the tasks with varying speeds and intensities. They also record the start times in a diary, which sometimes may not align with the actual task execution time.

The objective of this research is to develop an automated algorithm that integrates tap detection and movement analysis to accurately and efficiently identify the different activities performed by PD patients during the test battery. The algorithm leverages low-pass filtering, dynamic thresholding, and envelope-based movement detection to identify the various tasks. This research aims to eliminate the human influence during the annotation and efficiently identify the start and end times of the tasks. This development will reduce the influence of inter-observer variability and provide consistent and reliable monitoring of PD patients' activities.

Chapter 3

Methodology

3.1 Materials and Methods

3.1.1 Axivity AX6 Accelerometer Devices

PROPARK study uses the wearable Axivity AX6 [Axivity Ltd. (n.d.)], a 6-axis logging accelerometer device to monitor the movements as they seamlessly integrate into patients' daily routines, making it ideal for extended wear. Renowned for its accuracy, the AX6 consists of a tri-axial accelerometer and a tri-axial gyroscope, offering a high sampling rate crucial for nuanced movement analysis. Figure 3.1 shows the illustration of a sample Axivity device and the orientation of the accelerometer and gyro axis from the Technical Developer Guide, Axivity Ltd. (n.d.). Axivity Ltd. (n.d.) states that "The accelerometer sensor measures linear acceleration in each of the three axes (a_x, a_y, a_z) and the output is expressed as three-signed 16-bit integers. The gyroscope sensor output is expressed as three-signed 16-bit integers in the order x, y, z $(\omega_x, \omega_y, \omega_z)$. The orientation of the sensor axis is the same as the accelerometer. The output of the gyroscope is proportional to the angular velocity around each of the axes and is usually described in degrees per second."

Using the accelerometer we record the continuous data that can be used to track patients' movement patterns over time, which is crucial for evaluating the progression of Parkinson's Disease (PD) and the effectiveness of medicines, if any.

3.1.2 Low-pass filtering

In this study, a Butterworth low-pass filter was applied to the accelerometer data to reduce high-frequency noise that may obscure meaningful movement signals. The cut-off frequency for the filter was selected based on the expected frequency range of the patient's movements. The data were then passed through the filter before further analysis to ensure that only relevant movement data were retained.

Code Implementation:

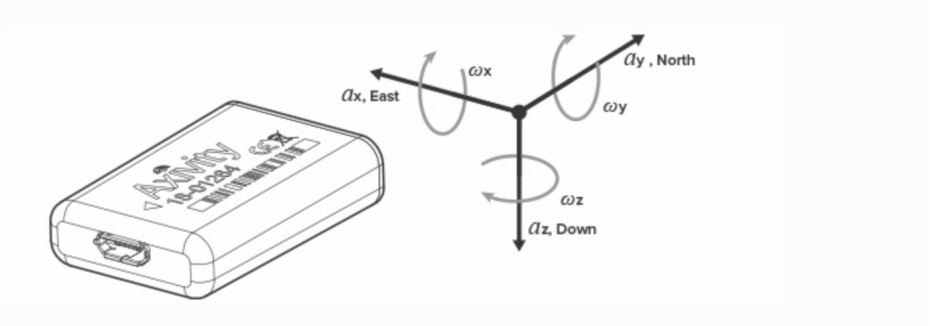


Figure 3.1: Illustration of Axivity device and the orientation of axes.

Using the Python **scipy.signal** module by SciPy Community (2008–2023) is one of the usual approaches to implement the Butterworth transfer function in code. Using the transfer function the coefficients are derived, which are then used to build a digital filter that efficiently reduces unwanted noise in a given signal. The **butter()** (SciPy Community — scipy.signal.butter (2008–2023)) and **filtfilt()** (SciPy Community — scipy.signal.filtfilt (2008–2023)) functions, for instance, are used in the construction of the Butterworth filter.

In this implementation, the filter is applied to the accelerometer data in order to remove high-frequency noise and retain the low-frequency movements. The use of the *scipy.signal.butter()* function allows the filter to be tailored for different cut-off frequencies based on the nature of the movements. *scipy.signal.filtfilt()* is then employed to apply this filter in a zero-phase manner, preserving the time alignment of the data. This ensures the integrity of the accelerometer signals while reducing any unwanted high-frequency components.

3.1.3 Signal Envelope Extraction

Envelope Using Standard Deviation

The standard deviation-based envelope extraction technique works very well for recording dynamic shifts, which makes it appropriate for data produced by accelerometers throughout a range of intensity activities. We modify this technique to extract the envelope by computing the standard deviation for the magnitude vector of the accelerometer signals x, y, and z-axes within a moving window of a certain size.
The moving window size, N, plays a crucial role in determining the level of sensitivity to amplitude changes. A smaller window size results in higher sensitivity to short-term variations, whereas a larger window size smooths out short-term fluctuations, focusing more on longer-term trends. We determine the window size to align with the specific dynamics of the task being evaluated, based on empiricial tetsing.
Formula:

For the magnitude of acceleration signals (M_n) along x, y, or z axes within a

moving window of size N, the standard deviation is computed using:

$$StdDev\_Envelope(M_n) = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(M_{n+k} - \bar{M}_n)^2} \tag{3.1}$$

where $\bar{M}_n$ is the mean of the magnitude values within the window.
For accelerometer magnitude, standard deviation-based envelope extraction is often chosen when:

- The intention is to highlight changes in the accelerative movements' amplitude.
- It is important to have a high sensitivity to sudden changes in acceleration.
- Non-stationary and dynamic motions, like walking, are common.

Envelope Using Variance

The variance-based envelope extraction technique works well for signals with extreme values or anomalies like gyroscopic signals during rotational motions. It is especially helpful when there are outliers present. By computing the variance for the angular velocity signal inside a moving window of a certain size, we modify this technique to extract the envelope.

The advantage of using variance over standard deviation lies in its ability to capture the spread of data around the mean, which can be useful in situations where accelerometer signals exhibit large swings or extreme values. This can occur, for example, during rotational motions or when signals are subject to sudden shifts. Variance-based envelope extraction ensures that the overall distribution of data, rather than just the magnitude of individual values.
Formula: For the angular velocity signal (W_n) along the z-axis within a moving window of size N, the variance is calculated using:

$$Variance\_Envelope(W_n) = \frac{1}{N}\sum_{k=1}^{N}(W_{n+k} - \bar{W}_n)^2 \tag{3.2}$$

where $\bar{W}_n$ is the mean of the angular velocity values within the window.
Variance-based envelope extraction is used when:

- It is necessary to have a reliable indicator of the distribution of angular velocity values, particularly while rotating.
- The item Signals with asymmetrical or non-Gaussian distributions are subject to the study.
- It is crucial to have resilience to extreme values, which are typical during gyroscopic signals during rotation.

The unique characteristics of the signal and the type of motion being studied determine whether to use variance-based or standard deviation-based envelope extraction techniques for accelerometer data. While applying variance to the gyroscope offers a robust characterization of angular velocity, it is especially useful during rotational motions. Using standard deviation for the magnitude of acceleration captures fluctuations in amplitude during dynamic activities. The choice of the moving window size, as mentioned earlier, is a critical factor influencing the efficacy of envelope extraction methods. We determine the window size based on the specific task that is being evaluated.

3.2 Data Collection

For this research, we use the dataset that has been collected during the PROPARK study. We have data from a total of 302 unique participants so far. We have 429 recordings, distributed across different follow-up points: baseline (262 recordings), 6-month follow-up (110), and 12-month follow-up (57). Some participants do not require follow-up recordings. Each participant contributes to a 7-day recording period, where they perform the set of eight motor tasks daily - once before dedication (OFF Test) and once after medication (ON Test), or an hour between assessments in case the patient is not taking medication. We also have a supervised recording known as Day 0, where participants perform the tasks under the guidance of a researcher at the hospital, where they learn the proper instructions to perform the battery test at home.

Each participant usually wears two Axivity AX6 sensors when performing the motor test - one on the wrist and one on the lower back. These devices record both accelerometer and gyroscope data. Participants maintain a diary where they record the start times of each task during the test. Before performing each task, the participants are instructed to tap the device three times to indicate the beginning of the task in the signal. A detailed explanation on how we filter the dataset based on diary data is provided in the section 3.2.2.

3.2.1 Instructions for the motor tasks

As part of the study, the participants perform the motor tests twice a day in the morning. Each test session consists of 8 tasks which together, take about 10 minutes to perform. Participants who are not yet taking Parkinson's medication perform the two test sessions one hour apart. For those on medication, the first session is performed immediately after waking up without medication, and the second session takes place one hour later after the participant has taken the medication. These sessions are referred to as the OFF (medication) and ON (medication) tests, respectively, allowing researchers to examine how mobility is affected by medication.

Participants are instructed to:

1. Write down the start time (in hours and minutes) of each task in their diary.
2. Set a timer for 30 seconds before each task to allow preparation and at least 20 seconds to complete the task.
3. Tap the accelerometer three times to indicate the start of the task.

The standardized motor test includes eight tasks, and the following are the instructions for each task:

Task 1: Finger Tapping

Step I: Open the page with the date on which the participant is completing the test session.

Step II: Sit in a comfortable chair.

Step III: Record the start time (in hours and minutes) of this assignment.

Step IV: Place the arm with the accelerometer on the top of your lap or a hard surface to stabilize it.

Step V: Set the timer for 30 seconds so that you have time to prepare for the test and at least 20 seconds to complete the task.

Step VI: Tap the accelerometer three times with your finger on your wrist to indicate the beginning of the task.

Step VII: Place the thumb of your other hand around the wristband of the accelerometer and place the other fingers in a natural way wrist.

Step VIII: Next, place your index finger just slightly above the accelerometer so you can tap with this finger.

Step IX: It is important that you only move your index finger during the assignment. Do not lift your arm.

Step X: Use your index finger to tap the accelerometer until the timer goes off (this must be at least 20 seconds).

Task 2: Resting while sitting

Step I: Remain seated comfortably in a chair.

Step II: Record the start time (in hours and minutes) of this assignment in your diary.

Step III: Set the timer for 30 seconds so that you have time to prepare for the test and at least 20 seconds to perform the task.

Step IV: Tap the accelerometer three times with your finger on your wrist to indicate the beginning of the task.

Step V: Place your hands on your thighs and hold your palms up.

Step VI: Close your eyes and count out loud from 100 until the timer goes off.

Task 3: Lifting arms

The steps I–IV are the same as the previous task.

Step V: Extend both arms straight in front of you. You may keep your palms down.

Step VI: Close your eyes and count out loud from 100 until the timer goes off.

Task 4: Hand open and closing

The steps I–IV are the same as the previous task.

Step V: Hold the arm on which you are wearing the accelerometer extended forward with your hand so that your fingers point up and your palm faces forward.

Step VI: Open and close your hand as fully and as quickly as possible.

Step VII: Repeat this movement until the timer goes off.

Task 5: Arm Rotation

The steps I–IV are the same as the previous task.

Step V: Stretch the arm on which you are wearing the accelerometer straight forward.

Step VI: Turn your palm alternately toward the ceiling and the floor, in a rotation motion.

Step VII: Repeat this movement as completely and quickly as possible until the timer goes off.

Task 6: Foot Stamping

The steps I–IV are the same as the previous task.

Step V: Grasp your knee with the hand you are wearing the accelerometer. If you wear the accelerometer on your right wrist, grab your right knee. If you wear the accelerometer on your left wrist, grab your left knee.

Step VI: While keeping your toes on the ground, raise the heel of the leg you are holding as high as possible and stomp the heel of that leg on the ground.

Step VII: Repeat this movement as completely and quickly as possible until the timer goes off.

Task 7: Stand up and walking

The steps I–IV are the same as the previous task.

Step V: Remain seated on the chair and cross your arms in front of your chest, with the arm with the accelerometer over the other arm.

Step VI: Stand up, pause briefly, and let the arms hang naturally along the body again.

Step VII: Walk forward in a straight line at a normal pace in a room where you can get the most distance without obstacles along the way.

Step VIII: At the end of the room, turn around and walk back to the chair.

Step IX: Repeat the walk until the timer goes off. Turn alternately left and right when you reach the end of the room.

Task 8: Standing still

The steps I–IV are the same as the previous task.

Step V: Stand up straight.

Step VI: Cross your arms, with the arm on which you are wearing the accelerometer over your other arm.

Step VII: Stand still and look straight ahead until the timer goes off.

3.2.2 Diary Data Filtering

At this stage in our experiments, we have worked with 358 wrist accelerometer data files. We have 381 unique PROPARK IDs and 379 Axivity IDs in the participant diary dataset. We rely on the participants' diaries, where they note the start times, as our reference points when aligning the accelerometer data. However, due to the manual nature of this task, several participants encounter challenges in accurately recording the time, and issues tend to arise, like variations in the data entry format. We have cleaned the dataset to follow a standardized date-time format. In particular, we used diary entries to filter the data for Task 1 and validate the availability of 15 tests, which included Day 0 and Day 1–7 OFF and ON tests. After this filtering, we have the dataset to include information from 251 participants' diary entries, which will be used as our reference for the start times. The histogram in Figure 3.2 shows the distribution of the non-empty values for tasks in the diary dataset. Here we can see that many participants have incomplete diary data, with entries for fewer than the full 15 tests (e.g., 14, 12 tests). Our analysis focuses on participants who have recorded all 15 sets of test data.

3.2.3 Data Annotation

Since the tasks are performed at home without supervision, data annotation becomes a challenge. The researchers manually annotate the start and end of the tasks for a participant. We have 15 manually annotated datasets, which serve as a Gold Standard for developing the algorithm, and to further validate and evaluate the performance of our algorithm.

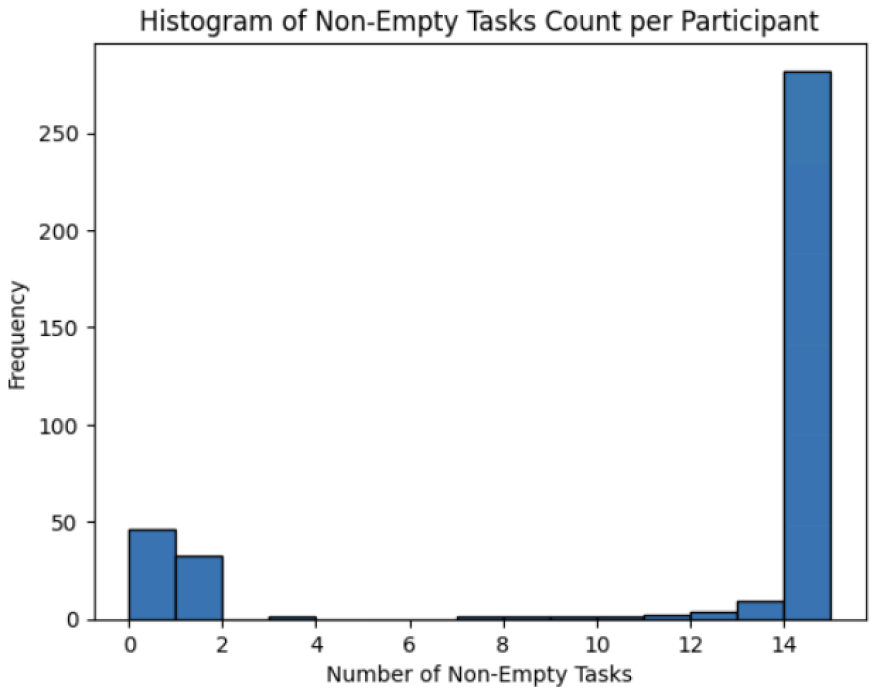


Figure 3.2: Histogram of the distribution of the non-empty tasks in the diary dataset.

3.3 Algorithm Development

The proposed algorithm consists of two main components:

1. **Tap Detection Module:** Utilizes signal processing techniques to identify distinct patterns indicative of three taps.

2. **Activity Task Marker Module:** Analyzes accelerometer data to recognize patterns associated with the start and end of standardized activity tasks.

3.3.1 Tap Detection Module

In the Tap Detection Module, the following steps were taken to identify the tapping events from the wrist accelerometer data.

1. **Data Input**

   - We process the 358 wrist accelerometer data (CWA) files and the diary dataset.
   - The CWA file consists of data collected each week, marking down the date in the file name along with their Axivity ID.

- We extract the Axivity ID and the corresponding PROPARK ID from the file names to uniquely identify each accelerometer dataset.

2. **Participant-specific Data Retrieval**

   - We then retrieve the diary data specific to each participant consisting of the start times for the tasks.

3. **Accelerometer Data Processing**

   - We utilize the CwaData package to read accelerometer sample values efficiently.
   - We have tried using a few packages available, but none of them worked as reliably as CwaData Package. Either they didn't read the CWA file fully or missed data.
   - We calculate the magnitude of the acceleration vector for the three axes (accel_x, accel_y, accel_z) within the sample values.

4. **Time Window Definition**

   - Using the diary entries, we acquire the task start date and time for all the tasks.
   - For Task 1, we create a time window ranging from one minute before the task's start date and time to 10 seconds after the next task's date and time.
   - For tasks 2 - 7, we create a time from 10 seconds before the task's start date and time to 10 seconds after the next task's date and time.
   - For task 8, we create a time window from one minute before the task's start date and time to one minute after that date and time.
   - We consider the human element here, as the participant might take some time to start the task or set up after performing the taps, so a 10-second buffer is used to account for this.

5. **Threshold Calculation**

   - We calculate a dynamic threshold for tap detection, set at 65 percent of the maximum magnitude within the specified task time window.
   - This is purely trial and error. We experimented with various thresholds, and 65% provided the ideal results. We can't set the threshold too high because then we might miss important peaks and information. If it's too low, we fail to accurately identify the specific three taps.
   - We found that setting the threshold at 65% of the maximum threshold ha given us the most consistent results across different test sessions. We were able to pick up the identifier three taps without being too sensitive to the

background task motion signals. This is clearly illustrated in the Figure 4.3, where we can see that the algorithm captures the three taps before Task 1.

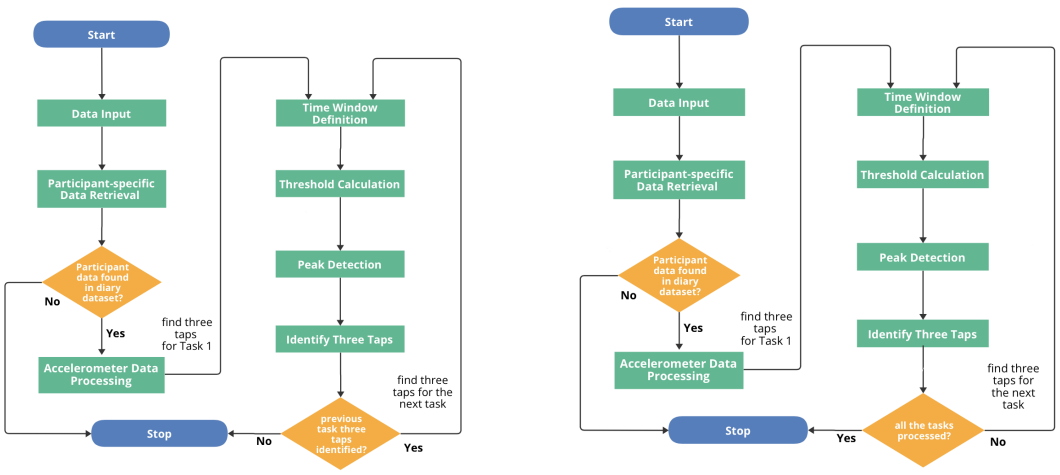
- By using a dynamic threshold, we can efficiently determine the peaks in the signal, which varies across tasks due to the differing movement intensities.

6. **Signal processing for Peak Detection**

   - We leverage the machine learning function *scipy.signal.**find_peaks*** to identify peaks in the magnitude signal that surpasses the calculated threshold within the window.

7. **Identify the Three Taps**

   - Locate the first three consecutive peaks within a 2-second interval.
   - Check for the existence of immediate peaks within the same 2-second window.
   - Verify and ensure a 3-second gap in both directions between the previous and next peaks. This ensures that only distinct tapping events are identified, not noise or multiple taps occurring too closely together.
   - Save Tap 1, Tap 2, Tap 3 times for each task when the above criteria are met.



(a) Sequential Tap detection module

(b) Independent Tap detection module

Figure 3.3: Flowcharts showing the architecture of Tap Detection Module

We have developed two versions of the algorithm. The architecture for the first version of the tap detection module is shown in the flowchart illustrated in Figure 3.3a.

In this version, called the sequential tap detection module, we aim to identify the three taps of a task if and only if the previous task's three taps are identified. For example, we proceed to identify three taps for task 3 only if three taps for task 2 are identified, and so on. This helps us understand the participant's performance throughout the test session, as the algorithm ensures the integrity of each task sequence. The architecture for the second version of the tap detection module is shown in the flowchart illustrated in Figure 3.3b. The second version called the independent tap detection module of the algorithm aims to identify all three taps of the tasks independently. That means the algorithm would proceed to identify three taps for Task 4 even if we do not find three taps for Task 3 in one test session. Here, we aim to identify all three taps for a participant so we can later use this information to identify the start and end times of the individual tasks. Both of these versions have resulted in very different results in terms of how many taps were detected. They both have their advantages, sequential detection ensures task order but misses identifying taps for the later tasks if earlier taps are missed. And independent detection improves tap detection for individual tasks. These trade-offs are further explored in Section4.1

The Tap Detection Module is aimed to robustly capturing and characterizing tapping events in the accelerometer data, even in the absence of perfect sequence completion, laying the groundwork for further analysis and algorithm refinement. Once the taps are detected, they can be used as soft anchors for defining task windows. For example, as illustrated in the Figures 4.7a and 4.7b, the signals that show high acceleration magnitude are three taps which are followed by the clear rise in signal activity. This can be used to accurately define the tasks' duration.

3.3.2 Activity Task Marker Module

In the Activity Task Marker Module, to determine the start and end times for each task, thresholds and analysis methods were carefully determined through trial and error for each task. The choice of data type (accelerometer vs. gyroscope) and statistical measures (standard deviation vs. variance) depends on the nature of the task. When rotational movements are involved, gyroscope data is used to capture angular velocity, while for non-rotational movements, accelerometer data and magnitude (combining accelerations from the X, Y, and Z axes) are used. For example, we can see the differences in the signal patterns in the Figures 4.7a and 4.9a. Standard deviation was used in the tasks involving more dynamic movement like finger tapping or hand opening and closing ((e.g., Task 1, Figure 4.7b), while variance helped in steady-state or stillness tasks, such as resting while sitting or standing still (Task 2, Figure 4.9d).

Additionally, thresholds for rise, drop, and "almost zero" are task-specific and were selected based on empirical testing and the specific movement patterns observed during each task across multiple test sessions. In the Figure 4.7b, we can notice how the envelope picks up where the activity begins and ends - which helped confirm these choices.

The following steps were taken to identify and mark the beginning and end of a

task.

1. **Data Preparation**

   - We load the participant data generated from the Tap Data module. This data is used because it has more comprehensive tap information for Task identification.
   - We Identify pairs of tasks where both Task 1 and Task 2 identifier taps are identified in the same test session.
   - Load the corresponding CWA file for each participant.

2. **Task 1: Finger Tapping**

   The following steps explain how we analyze the envelope data generated by accelerometer data to determine the start and end timings of Task 1, taking into account certain threshold values for scenarios such as rise, drop, and almost zero.

   (a) Create Task Window

   - For each identified pair of tasks:
     - We create a window using the timestamp of Task 1, Tap 1, and Task 2, Tap 1 using the participant data.
     - We then create a subset of the accelerometer data within this window.

   (b) Calculate magnitude vector

   - We calculate the magnitude vector of the subset accelerometer data.
   - Magnitude is computed using the accelerometer data, as this task does not involve rotational movement.

   (c) Generate Envelope

   - We then generate an envelope by calculating the standard deviation within a moving window of size 100 for the magnitude vector values.

   (d) Set Threshold values

   - Set threshold values for rise, drop, and almost zero in the envelope:
     - Threshold Rise: 0.05
     - Threshold Drop: 0.02
     - Threshold Almost Zero: 0.01
     - Duration: 10 seconds
   - Thresholds were determined through empirical testing.

   (e) Identify start time

- We find the start time of Task 1 by checking for a continuous rise in the envelope data points. The rise threshold starts at 0.05.
- If within 10 seconds a drop in the data points occurs below the threshold drop value (0.02), find the next continuous rise pattern.
- We categorize the start time of the task at the timestamp where we encounter the first data point that satisfies the above criteria.

(f) Identify end time

- If and only if the start time is found:
  - Find the first data point where the threshold is less than 0.01 and the next five indices are also less than 0.01.
  - If the next rise is immediate (within a 2-second interval), continue finding the next almost zero index, which is considered the end of the task.

3. **Task 2: Resting while sitting** and **Task 3: Lifting arms**
The following steps explain how we analyze the envelope data generated by accelerometer data to determine the start and end timings of Task 2 and Task 3, taking into account certain threshold values for scenarios such as rise, drop, and almost zero.

(a) Create Time Window

- For Task 2, we create a window using the timestamp of Task 2, Tap 1, and Task 3, Tap 1 using the participant data. And for task 3, we create a window using the Task 3, Tap 1 and Task 4, Tap 1 timestamps.
- We then create a subset of the accelerometer data within this window.

(b) Low-pass filtering

- We pass the subset of accelerometer data through a low-pass filter to remove noise. We use the Butterworth filter.
- From the filtered data, we use the gyroscope Z-axis(gyro_z) to further analyze the movement.
- We choose to use gyro_z for the envelope analysis because this task involves rotational movements, which occur around the forearm's longitudinal axis.

(c) Generate Envelope

- Generate an envelope by calculating the variance within a moving window of size 100 for the gyro_z values.
- For this task, we use angular velocity from the gyroscope data, as the movement involves significant rotational changes. The variance of the angular velocity is calculated.

(d) Set Threshold Values

- Deviation threshold: 0.1
- Duration: 10 seconds.

(e) Identify Start time

- The timestamp when the variance drops below the threshold and remains stable for at least 10 seconds is classified as the start time.

(f) Identify End time

- The first timestamp after the start time when the variance rises above the threshold is classified as the endpoint.

4. **Task 4: Hand open and closing**
We follow the same steps as Task 1 with two main differences.

   - We create a window using the timestamp of Task 4, Tap 1, and Task 5, Tap 1 using the participant data.
   - Steps (b) - (e) are the same as mentioned in Task 1.
   - We find the first data point after the start time where the threshold is less than 0.01 and classify it as the end time.

5. **Task 5: Arm Rotation** We follow the same steps as Task 2 with varying threshold values.

   - We create a window using the timestamp of Task 5, Tap 1, and Task 6, Tap 1 using the participant data.
   - Threshold values:
     - Threshold rise: 50
     - Threshold drop: 20
     - Threshold almost zero: 50

6. **Task 6: Foot Stamping** We follow the same steps as task 4 with varying threshold values.

   - We create a window using the timestamp of Task 6, Tap 1, and Task 7, Tap 1 using the participant data.
   - Threshold values:
     - Threshold rise: 0.1
     - Threshold drop: 0.05
     - Threshold almost zero: 0.1

7. **Task 7: Stand up and Walking** We follow the same steps as Task 1 up until step (d).

- We create a window using the timestamp of Task 7, Tap 1, and Task 8, Tap 1 using the participant data.
- We find the start time of Task 7 by checking for a continuous rise in the envelope for at least 10 seconds. If we encounter a drop in the data point below the threshold drop value within 10 seconds, we check if there is an immediate rise in the envelope value above the rise threshold within 2 seconds. If yes, we continue to check this pattern and categorize this as the start time.
- We find the first data point after the start time where the threshold is less than 0.1 and classify it as the end time.

8. **Task 8: Standing still**

    - We create a window using the timestamp of Task 8, Tap 1, using the participant data and one minute.
    - We follow all the same steps as Task 2 as it is a movement that involves finding the pattern where the participant is being still.

All the methodological choices - from selecting the signal types to setting the thresholds for each task were determined by analysing the signals for the individual tasks during data exploration. The next chapter shows how the algorithm performed across different tasks for various patients and how these design choices held up across various patient data.

Chapter 4

Results

4.1 Tap detection Module

This section presents the results from the tap detection module across various tasks from 251 patients. We compare two versions of the tap detection system—sequential and independent—highlighting the total number of taps detected, average time differences, and their respective distributions. The plot in Figure 4.1 shows an example of

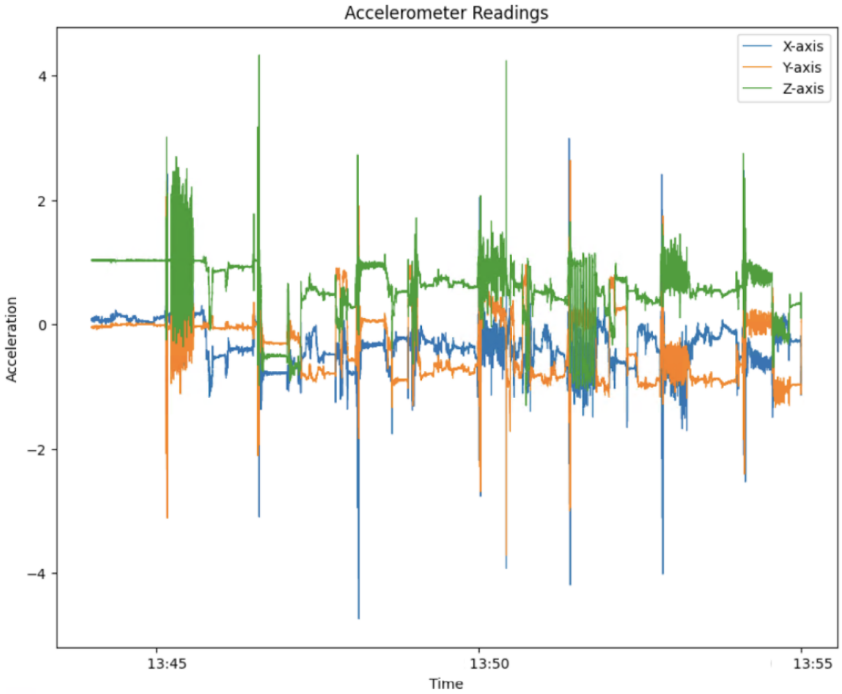


Figure 4.1: Plot showing the acceleration of X, Y, and Z-axes from the accelerometer data.

the accelerometer data from one test session performed by the participant. It shows

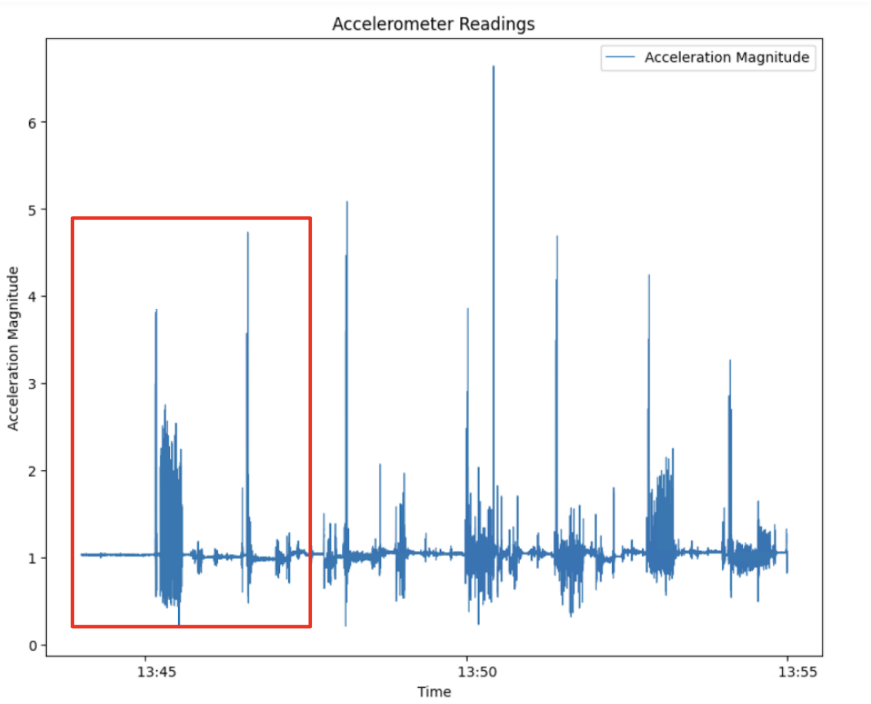


Figure 4.2: Plot showing the acceleration magnitude vector of X, Y, and Z-axes from the accelerometer data.

the acceleration X, Y, and Z-axes signals vs. Time. Typically, when a participant performs the test session for around 10 minutes, which is shown here. We see the distinct spikes in the signal, which is indicative of the three taps on the accelerometer to indicate the beginning of a task, which we aim to identify using the algorithm. This is one of the instances where the participant performs all eight tasks as per instructions successfully. The plot in Figure 4.2 shows the acceleration magnitude vector of the three axes for one test session. The acceleration magnitude generates a single value that indicates the total intensity of motion, regardless of direction. We are interested in identifying the tap patterns rather than precise direction. Thus, using a single magnitude variable simplifies the process instead of analyzing X, Y, and Z axes individually. Typically, taps result in high acceleration magnitudes, which makes threshold-based detection more straightforward. Also, the magnitude takes into account the overall effect of all components, it can help lessen the impact of noise if there is noise or slight changes in the individual acceleration components. In Figure 4.2 we observe sharp peaks in the beginning of the session that are likely to be the three identifier taps. These taps stand distinctly from the rest of the tasks' activity. This supports our choice to use acceleration magnitude instead of analysising the X,Y and Z- axes individually. We consider each task individually to set and calculate the threshold: 65% of the maximum magnitude. As mentioned earlier in methodology, this threshold was determined through trial and error, allowing for the reliable iden-

tification of taps while minimizing noise from minor accelerations that could occur during the task. Subsequent steps include identifying all the peaks above this threshold to further identify the three taps. Figure 4.3 shows the time window that we consider for task 1, in this instance. The red dots characterize all the peaks within this window above the threshold. The algorithm identifies and saves the time stamps of the three taps. Figure 4.4 corresponds to the moments when taps were detected.

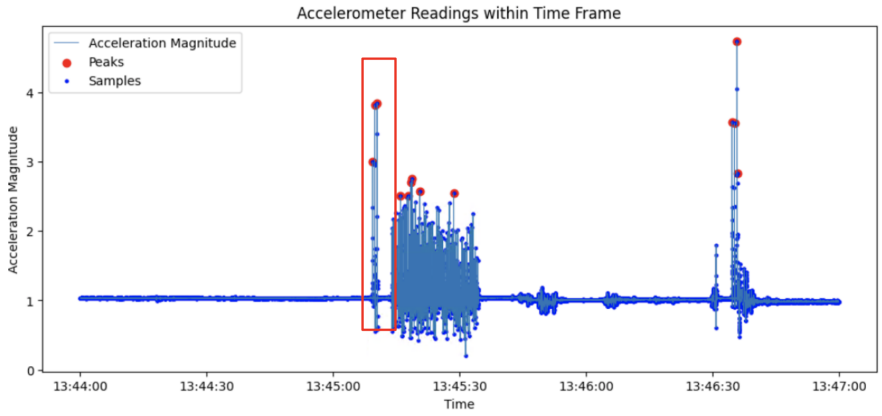


Figure 4.3: Plot showing the Time window considered for identifying three taps for task 1

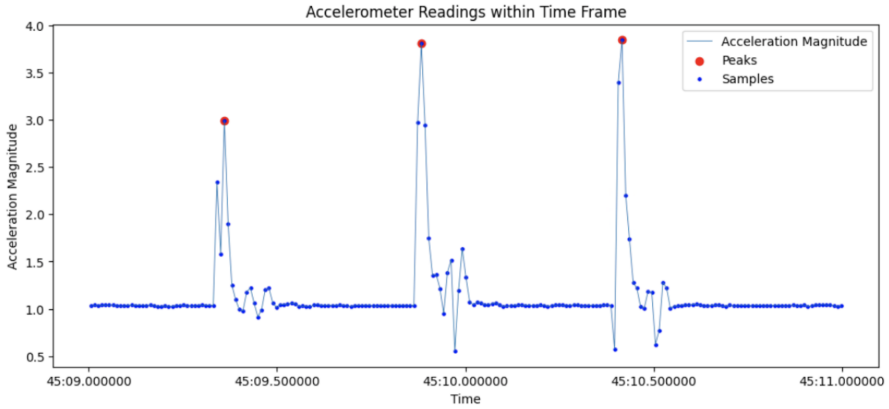


Figure 4.4: Plot showing zoomed-in view of the three peaks.

Tap Detection Analysis: Sequential vs Independent Methods

We have explored two logical approaches for Tap Detection module. Sequential - processes one full test session from Task 1 to Taksk 8 and Independent - processes each task on it's own. The two detection methods yield distinct tap counts and time differences, as summarized in Tables 4.1 and 4.2.

Table 4.1: Table showing the overall Tap count for Sequential Tap detection module

Task	Total Task Taps Identified	Average Time Difference (s)	Standard Deviation (s)
Task 1	1334	36.94	33.20
Task 2	552	23.79	19.82
Task 3	292	25.88	19.67
Task 4	158	25.92	17.65
Task 5	87	26.43	20.92
Task 6	38	25.28	18.27
Task 7	23	17.21	12.82
Task 8	16	15.81	11.99

As mentioned earlier, we use the diary time as our reference to indicate the beginning of a task. Comparing the diary time and the timestamp of the first detected 'tap', we calculate the time difference (in seconds). This computation offers insightful information on the temporal aspects of tap detection. The average time difference provides insight into the responsiveness of the tap detection system, which is crucial for determining its usability. Typically, the participant writes notes the time in their diary, sets the timer to perform the task, and then proceeds to perform the three taps followed by the actual task, so a slight delay is expected. The column "Average Time Difference" shows the average amount of time that elapses between the time that was entered in the diary and the time that the tapping was first noticed. This measure gives us an idea of how well the tap detection system aligns on average with the timing that is predicted using the diary time. The "Standard Deviation" column show hoe much the time difference varies across tapping times. A high standard deviation implies that there is more variation between the times when taps were detected and the diary time. A low standard deviation means that the detected taps are more aligned with the diary times. This could be due to the differences in how the participants record their diary times or the variations in the algorithm detection due to intensity of the taps.

The sequential method detects the taps only if the previous task's taps have been identified. We aim to recognize the taps for all the tasks in one complete session. However, it comes at the cost of fewer taps being recognized. In Table 4.1, looking at the total number of taps identified in the "Total Task Taps Identified" column, we notice that there is a clear downward trend. Task 1 starts with 1334 taps but by Task 3 it has already dropped to 292 taps. And by Task 8 only 16 taps are identified. This shows that as the session progresses, the algorithm recognizes fewer and fewer taps. The idea behind the sequential method was to find a full session of taps per patient, but it seems too restrictive. In this method, once a tap is missed for an earlier task, the algorithm stops looking to identify the taps in the following task. This is causing a significant loss in data. The "Average Time Difference" column tells us when the taps occurred compared to the time that the patient has noted in their diary. For Task 1, this time difference is around 36.96 seconds. It means that the patient has

taken some time to start the task after noting down the time. As tasks go on, the time difference decreases. This could mean that the patients find the rhythm and ease in the transition between the tasks. The "Standard Deviation" column measures how consistent the time differences are among various patients. In Task 1, the standard deviation is 33.20 seconds, which is slightly high. It means that some patients have taken more time than the diary time to start the taps, whereas some patients have tapped closer to it. By Task 8, the standard deviation has reduced to 11.99 seconds. It could mean that the taps are more consistent. However, towards the end, very few taps are detected and we may be working with a biased sample by this point. This method may be useful to find at least one full session per patient, but is less useful for analyzing individual tasks.

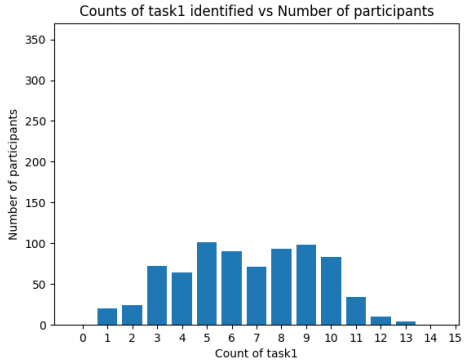
Figure 4.5 illustrates the distribution of the total number of taps identified across patients for each task using the Sequential Tap Detection module. Each histogram represents the number of participants (Y-axis) with a specific number of identified taps (X-axis) for a given task. These histograms help us validate our table-based observations from Table 4.1 visually, showing the fall-off in tap detection performance across tasks.

- As shown in Figure 4.5a (Task 1 Distribution) and Figure 4.5b (Task 2 Distribution), the tap detection algorithm performs reasonably well in the initial tasks. Most patients had 5-10 detected taps, indicating that the algorithm had identified their taps in a large portion of the trials. A small number of participants had 0 taps detected. This will affect the tap detection in the later tasks.

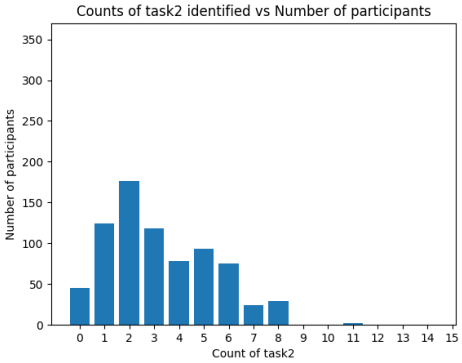
- In Figure 4.5c (Task 3 Distribution), Figure 4.5d (Task 4 Distribution), and Figure 4.5e (Task 5 Distribution), we notice a clear downward trend. The distribution is left-skewed with most patients having a 0-3 tap detection count and fewer patients having high tap counts. This trend can be observed in Table 4.1, where the total detected taps for Task 3 drop significantly (292 taps) compared to Task 1 (1334 taps).

- In Figure 4.5f (Task 6 Distribution), Figure 4.5g (Task 7 Distribution), and Figure 4.5h (Task 8 Distribution), the number of detected taps is drastically reduced. The majority of the participants have 0 taps detected. This suggests that because of the sequential interdependency in detection, the algorithm misses identifying taps for all the later tasks. This confirms what we observed in Table 4.1, where the tap count falls to just 38 for Task 6 and 16 for Task 8.

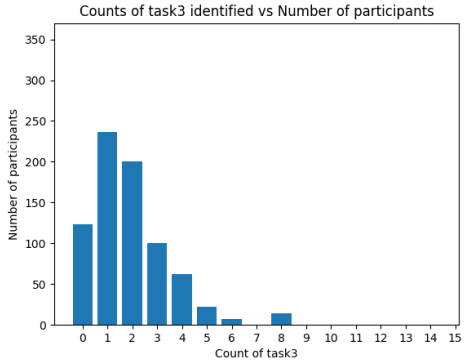
The independent method detects the taps for every task regardless of whether the previous task has been identified or not. This means that even if the taps are missed for Task 1, the algorithm still looks to identify the taps for Task 2 and so on. This approach yields in a higher number of taps identified across all tasks. In Table 4.2, looking at the "Total Task Taps Identified" column, right away we notice that the total number of taps detected across all the tasks is significantly higher than



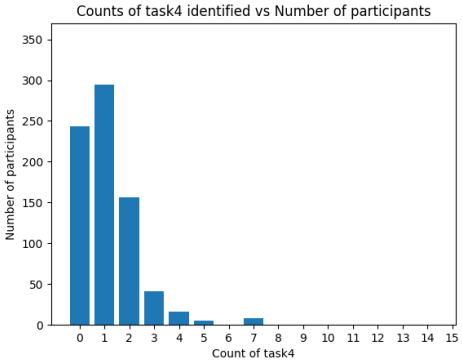
(a) Task 1 Distribution



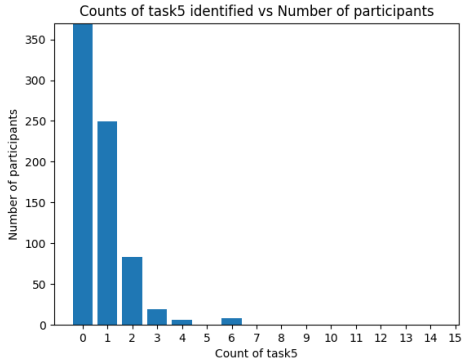
(b) Task 2 Distribution



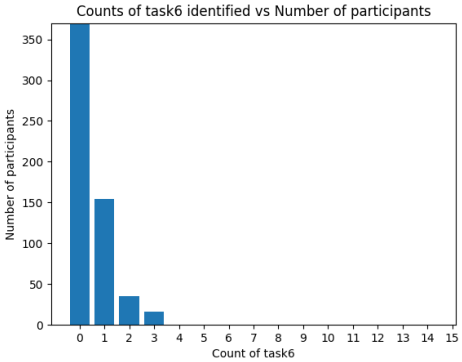
(c) Task 3 Distribution



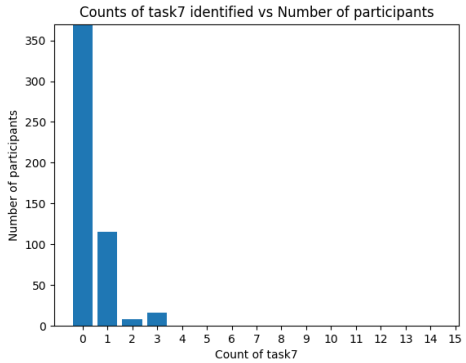
(d) Task 4 Distribution



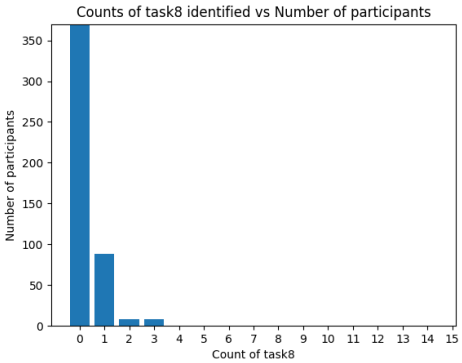
(e) Task 5 Distribution



(f) Task 6 Distribution



(g) Task 7 Distribution



(h) Task 8 Distribution

Figure 4.5: Distribution of the number of taps identified across various tasks for the participants - sequential tap detection module

Table 4.2: Table showing the overall Tap count for Independent Tap detection module

Task	Total Task Taps Identified	Average Time Difference (s)	Standard Deviation (s)
Task 1	1225	37.39	33.90
Task 2	1323	21.04	25.30
Task 3	1668	27.53	32.48
Task 4	1550	24.78	33.78
Task 5	1428	20.12	30.19
Task 6	1092	30.06	34.49
Task 7	1201	19.19	34.37
Task 8	1561	17.95	13.48

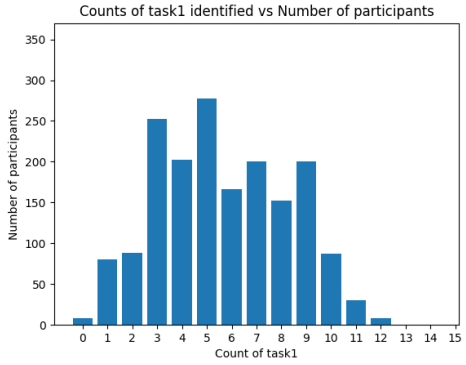
the sequential method. Task 1 has 1225 taps and Task 8 has 1561 taps. This just shows that using this method, we can identify the taps across the later tasks. The Average time difference for Task 1 is 37.39 seconds, but by Task 8 it is 17.95 seconds. This suggests that the detection is fairly reliable and stable throughout the session. Looking at the standard deviation, we can note that the detection taps are consistently aligned with the diary times. This method is better for analyzing the individual tasks as it captures more taps, reduces variability, and remains fairly stable across all the tasks.

Figure 4.6 illustrates the distribution of the total number of taps identified across patients for each task using the Independent Tap Detection module. Each histogram represents the number of participants (Y-axis) with a specific number of identified taps (X-axis) for a given task. Unlike the Sequential Tap Detection Module, this method processes each task independently. Comparing this figure with Figure 4.5 (Sequential Method) and Table 4.2, we can observe a significant improvement in tap detection stability and consistency across tasks.

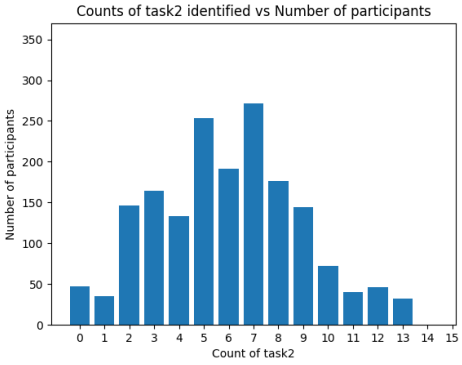
- As seen in Figure 4.6a (Task 1 Distribution) and Figure 4.6b (Task 2 Distribution), the majority of patients fall within the 5–10 tap range. Very few participants have 0 detected taps. This means that the algorithm correctly identifies the taps for most of the patients for these tasks.

- In Figure 4.6c (Task 3 Distribution), Figure 4.6d (Task 4 Distribution), and Figure 4.6e (Task 5 Distribution), we see that patients still have 5–10 detected taps in most cases. This means that the tap detection is relatively stable for these tasks.

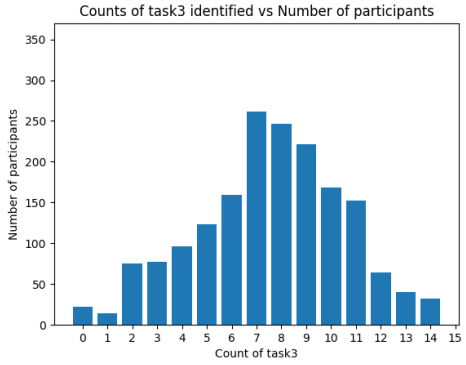
- n Figure 4.6f (Task 6 Distribution), Figure 4.6g (Task 7 Distribution), and Figure 4.6h (Task 8 Distribution), tap detection remains strong for most participants. These observations align with our observations from the Table 4.2.



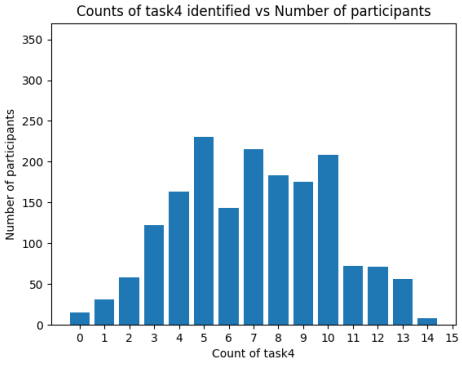
(a) Task 1 Distribution



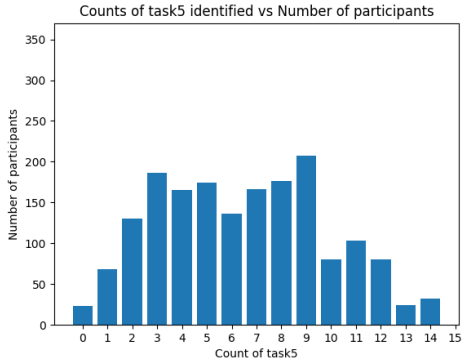
(b) Task 2 Distribution



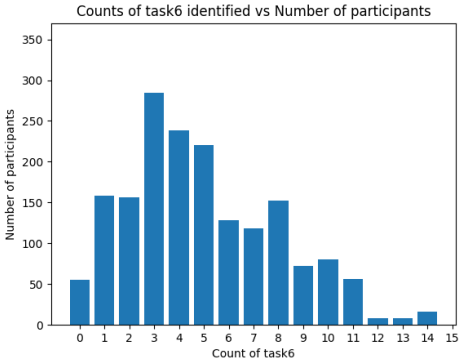
(c) Task 3 Distribution



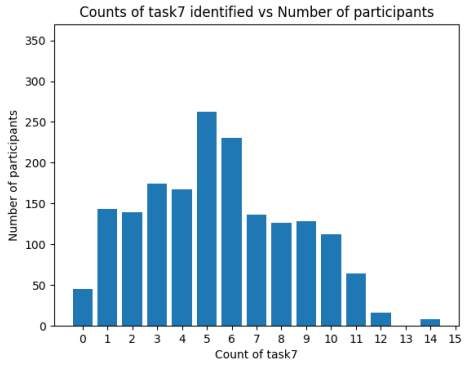
(d) Task 4 Distribution



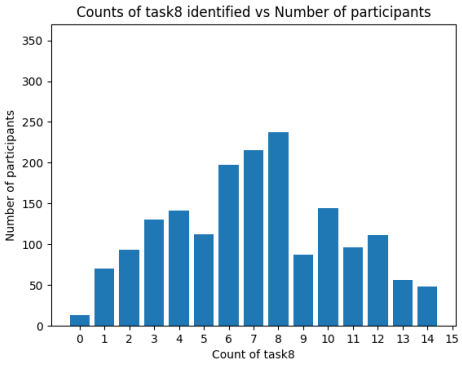
(e) Task 5 Distribution



(f) Task 6 Distribution



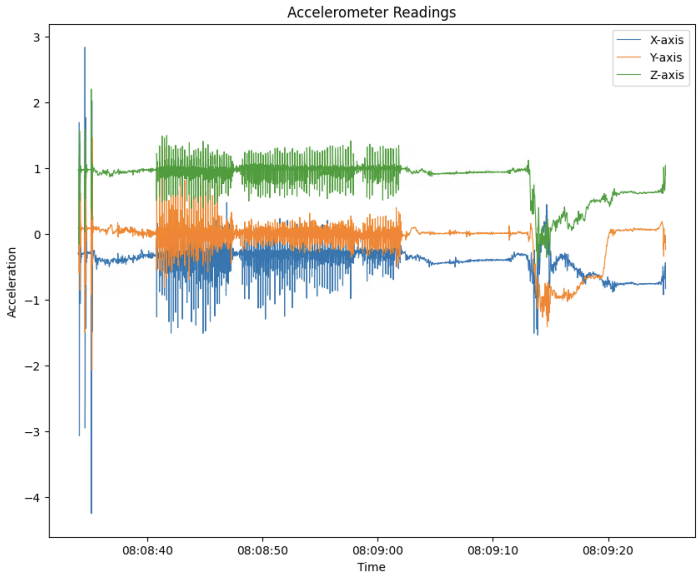
(g) Task 7 Distribution



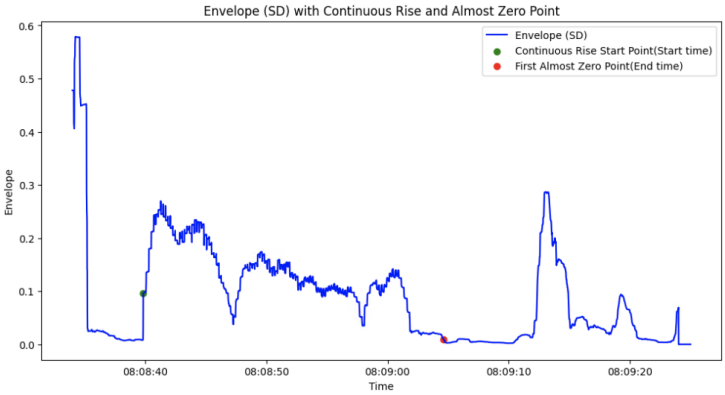
(h) Task 8 Distribution

Figure 4.6: Distribution of the number of taps identified across various tasks for the participants - independent tap detection module

4.2 Activity Task Marker Module



(a) Plot showing the acceleration of X, Y, and Z-axes from the accelerometer data for Task 1 within the given time frame.



(b) Plot showing the envelope of the accelerometer data, start (green marker) and end (red marker) times of Task 1

Figure 4.7: Task 1: Acceleration and Start & End Time Markers

In this section, we illustrate the process of identifying the start and end times of tasks using the Activity Task Marker module. For each task, we present accelerometer data along the X, Y, and Z axes within a defined time window. This time window enables the identification of start and end times for a given task.

4.2.1 Task 1 - Finger Tapping

Task 1 involves tapping the accelerometer with the patient's index finger. The following plots illustrate two examples of the accelerometer readings recorded during this task, highlighting the motion patterns and the start/end detection process.

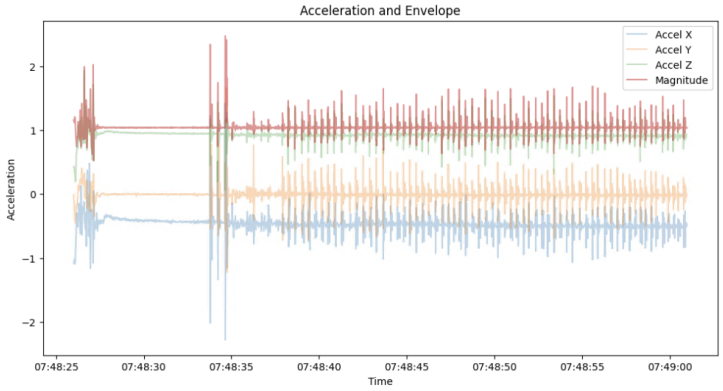
Accelerometer Readings

Example 1: The plot in Figure 4.7a illustrates the acceleration data along the X, Y, and Z axes during Task 1. It captures all the fluctuations from the participant's movements. The initial three sharp peaks in all three axes are the indicator taps that the patient performs to signify the beginning of a task. It is followed by periodic fluctuations representing the actual tapping phase. Since this task focuses on the finger movement and not on arm or wrist movement, we notice that the Y-axis shows more fluctuations signifying the vertical movement on the accelerometer than the X and Y axes.

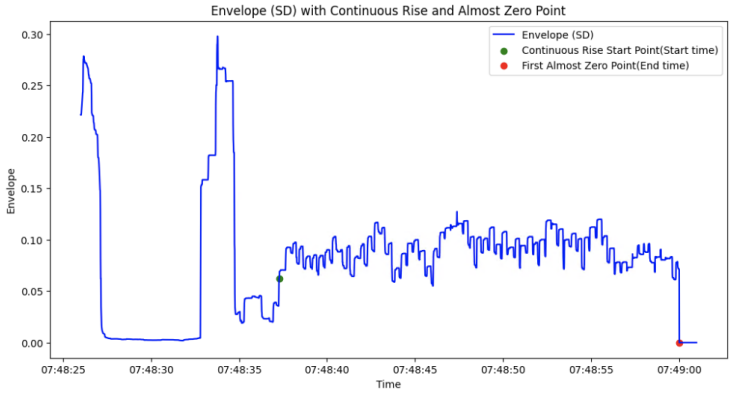
Task 1 Detection Using Standard Deviation Envelope

We chose to use the Standard Deviation of the acceleration magnitude as the envelope for Task 1 because it captures the intensity of the rapid finger movements rapidly. Figure 4.7b visualizes the envelope of the accelerometer data, which shows an overview of the activity intensity. The rising phase of the envelope curve corresponds to the active tapping and the curve as it reaches near zero corresponds to the task completion. The task start time is marked by a green marker, and it is identified by detecting "continuous rise" threshold point in the envelope. The red marker represents the task end time, determined by the "first almost zero point" using the predefined threshold where the envelope stabilizes after the last tap.

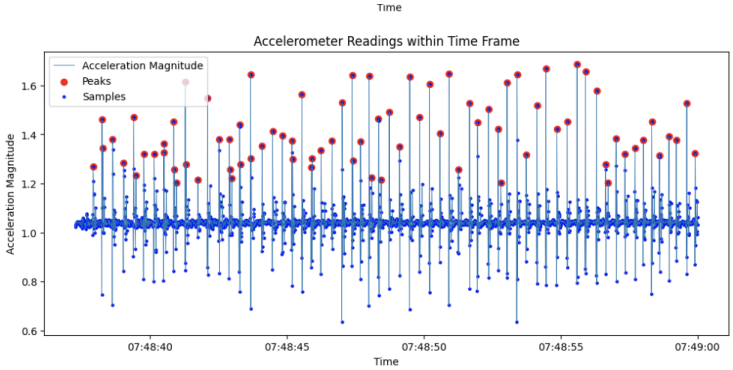
Example 2: In this example, we present three plots to provide an overall flow for detecting start and end times for Task 1. Figure 4.8a illustrates the acceleration data along the X, Y, and Z axes, along with the magnitude curve. The magnitude curve offers insights into the overall movement pattern. It shows periodic peaks that align with each individual tap. Figure 4.8b shows the envelope of the accelerometer data. It also shows the start and end times, green and red marker, which are identified at 2023-03-24 07:48:37.272 and 2023-03-24 07:49:00.016 timestamps respectively. For this particular patient, using the identified start and end times, we can calculate that it has taken approximately 22.744 seconds to complete Task 1. In Figure 4.8c, we can see a total of 79 peaks within the identified task duration. This indicates the total number of finger taps performed during this task. By analyzing the total number of taps performed during a task, the researchers can further monitor and compare the patient's performance across different sessions. They can also track potential improvements or deterioration over time.



(a) Plot showing the acceleration of X, Y, and Z-axes and magnitude for Task 1 within the given time frame.

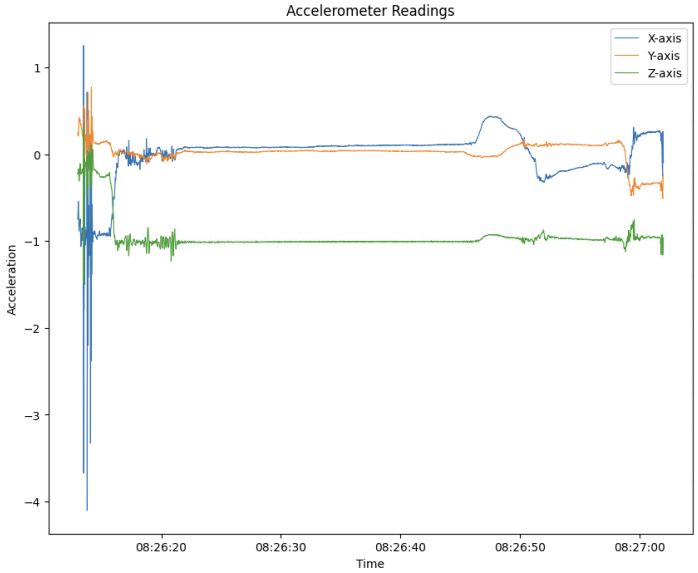


(b) Plot showing the envelope of the accelerometer data, start (green marker) and end (red marker) times of Task 1

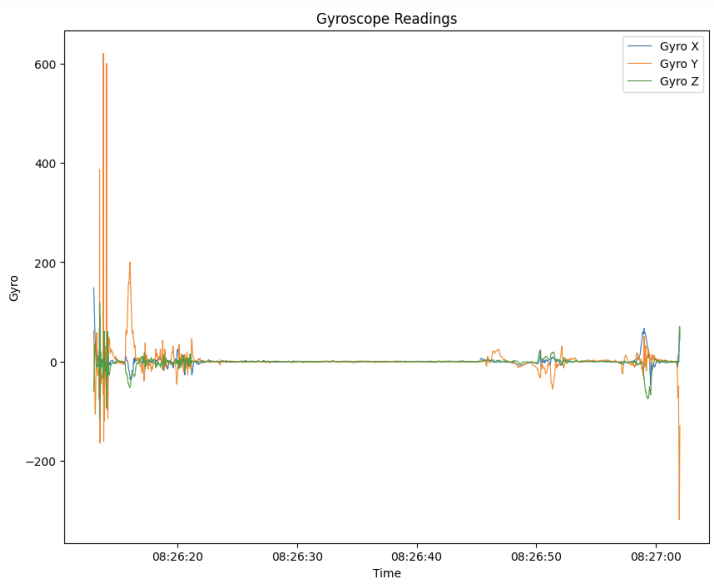


(c) Zoomed in plot showing the number of taps performed by the patient during Task 1.

Figure 4.8: Task 1, Example 2: Acceleration and Start & End Time Markers

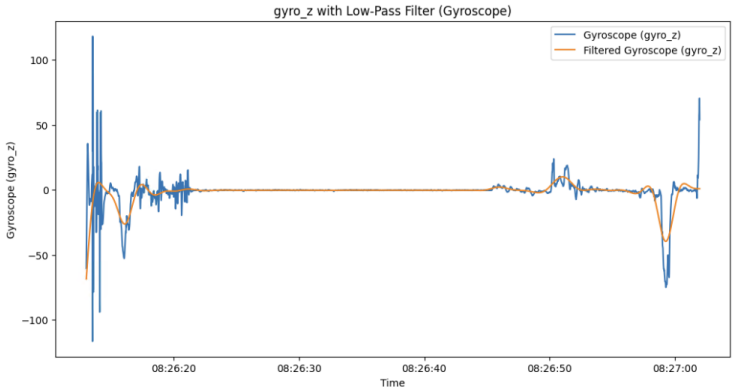


(a) Plot showing the acceleration of X, Y, and Z-axes from the accelerometer data for Task 2 within the given time frame.

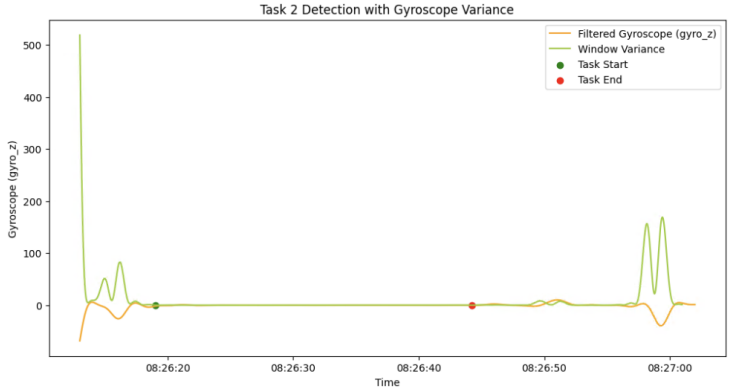


(b) Plot showing the gyroscope data of X, Y, and Z-axes for Task 2 within the given time frame.

Figure 4.9: Task 2: Acceleration and Gyroscope Data



(c) Plot showing the gyroscope of Z-axis before and after applying the filter for Task 2 within the given time frame.



(d) Plot showing the filtered gyroscope of Z-axis, envelope generated using window variance, start (green marker) and end (red marker) for Task 2.

Figure 4.9: Task 2 (continued): Start & End Time Markers

4.2.2 Task 2 - Resting while sitting

Task 2 involves resting both arms on the thighs while maintaining minimal movement. The following plots illustrate the accelerometer and gyroscope readings recorded during this task highlighting the motion patterns, filtering effects, and the start/end detection process.

Accelerometer Readings

Figure 4.9a visualizes the acceleration data for the X, Y, and Z axes during Task 2. Initially, there is a noticeable spike in all three axes indicating the three taps on the accelerometer to mark the beginning of the task. These taps serve as a manual marker to differentiate between consecutive tasks. Following the spikes, the acceleration stabilizes reflecting on the patient's minimal movement while sitting. The X and Y axes

remain close to Zero, indicating a stable posture and minimal movement, while the Z-axis shows a negative value representing the gravity acting on the accelerometer.

Gyroscope Readings

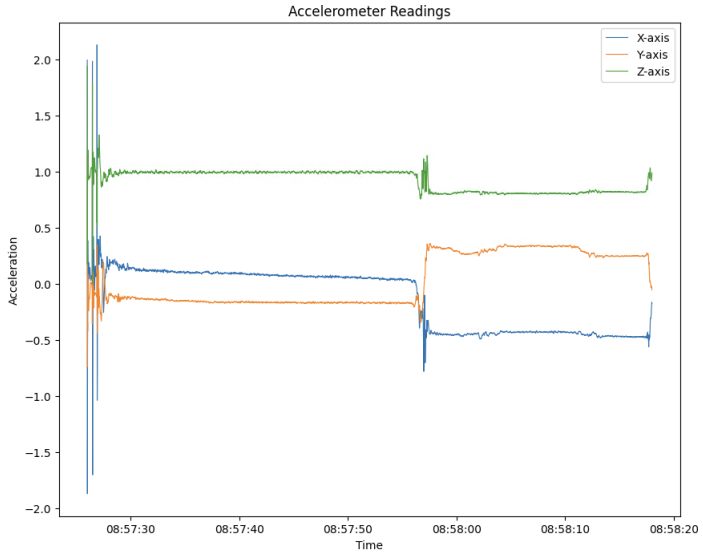
Figure 4.9b displays the angular velocity of the X, Y, and Z axes. Similar to acceleration data, there is a significant fluctuation in the readings indicating the initial three taps to signal the transition to this task. After this, all three gyroscope readings are close to zero indicating that the patient remained still throughout the task. Small fluctuations can be seen, which could indicate any involuntary movements by the patient like breathing or adjusting the hands.

Filtered Gyroscope Z-Axis

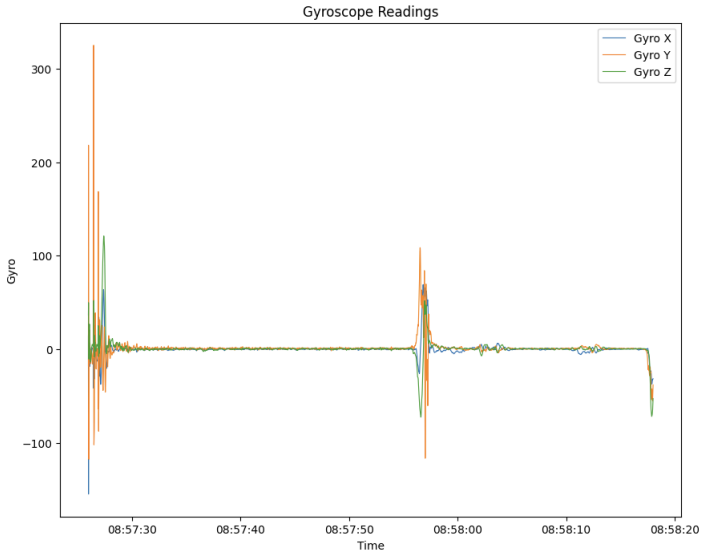
Figure 4.9c highlights the effect of applying a low-pass Butterworth filter on the Z-axis gyroscope data. Notice that there is some noise in the original raw signal, which is eliminated to retain only relevant information. After filtering, the gyroscope Z-axis data becomes much smoother, which helps detect the significant motion patterns while eliminating the random and minor fluctuations.

Task 2 Detection Using Variance-Based Envelope

We chose to use Variance-based envelope for this task because the variance stays low during the stillness periods and raises with any small movements.Figure 4.9d visualizes the filtered gyroscope Z-axis data along with a variance-based envelope to detect activity within the task duration. The filtered gyro_z data (orange) is plotted alongside the window variance (green), which captures significant motion variations. As mentioned in the methodology, there is a predefined variance threshold (0.1); to ensure only meaningful changes are detected, and a stability duration of 10 seconds; to ensure the detection is not triggered by short fluctuations. To detect the start and end of the resting phase, we used a variance-based envelope on the filtered gyroscope Z-axis data. The green marker indicates the detected start time, when variance is stabilized below the threshold and the red marker marks the first significant increase in variance, which signifies the end of the resting period. The variance envelope highlights the movements that are prominent in the beginning because of the initial taps and remain low during the task duration. Toward the end, there is a slight movement likely indicating the patient's movements as they conclude the task.

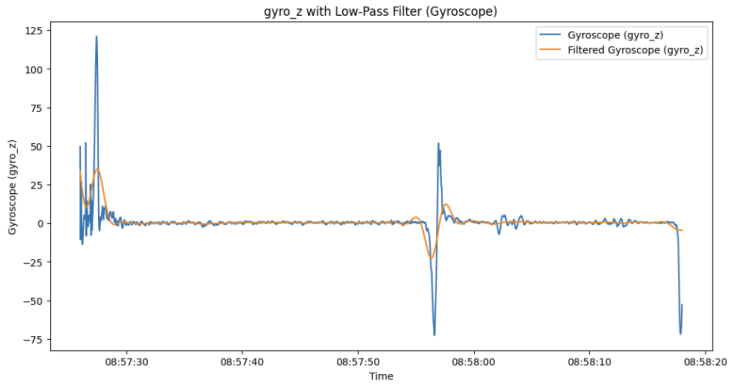


(a) Plot showing the acceleration of X, Y, and Z-axes from the accelerometer data for Task 3 within the given time frame.

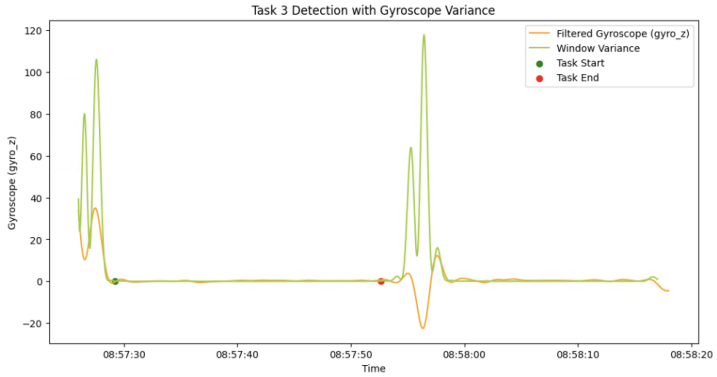


(b) Plot showing the gyroscope data of X, Y, and Z-axes for Task 3 within the given time frame.

Figure 4.10: Task 3: Acceleration and Gyroscope Data



(c) Plot showing the gyroscope of the Z-axis before and after applying the filter for Task 3 within the given time frame.



(d) Plot showing the filtered gyroscope of the Z-axis, envelope generated using window variance, start (green marker) and end (red marker) times for Task 3.

Figure 4.10: Task 3 (continued): Start & End Time Markers

4.2.3 Task 3 - Lifting Arms

Task 3 involves putting both arms straight in front while keeping the palms downward and maintaining minimal movement. The following plots illustrate the accelerometer and gyroscope readings recorded during this task highlighting the motion patterns, filtering effects, and the start/end detection process.

Accelerometer Readings

Figure 4.10a illustrates the acceleration data across the X, Y, and Z axes during Task 3. Similar to Task 2, the initial spike in the acceleration indicates the three taps identifying the beginning of the task. We can observe the moment the arms are raised from the changes in the X and Y axes. The X-axis shifts negatively while the Y-axis shifts, which indicates the forward movement of the arms. The Z-axis remains

relatively stable, showing the effect of gravity on the accelerometer when the arms are lifted.

Gyroscope Readings

Figure 4.10b visualizes the angular velocity of X, Y, and Z axes during Task 3. The three sharp peaks at the beginning of the time window indicate the task initiation. The next peaks indicate the subsequent spike in angular velocity around the time that the arms are lifted, which aligns with the expected rotational movement of the arms. After this, the gyroscope readings remain close to zero, indicating that there's no rotational movement in the arms while holding the position. The small fluctuation in the readings towards the end indicates the patient lowering their arms to mark the end of Task 3.

Filtered Gyroscope Z-Axis

Figure 4.10c compares the raw and filtered gyroscope Z-axis data. To reduce the noise in the raw data, we apply Butterworth low-pass filter. The signal becomes smoother and makes it easier to identify meaningful motion patterns.

Task 3 Detection Using Variance-Based Envelope

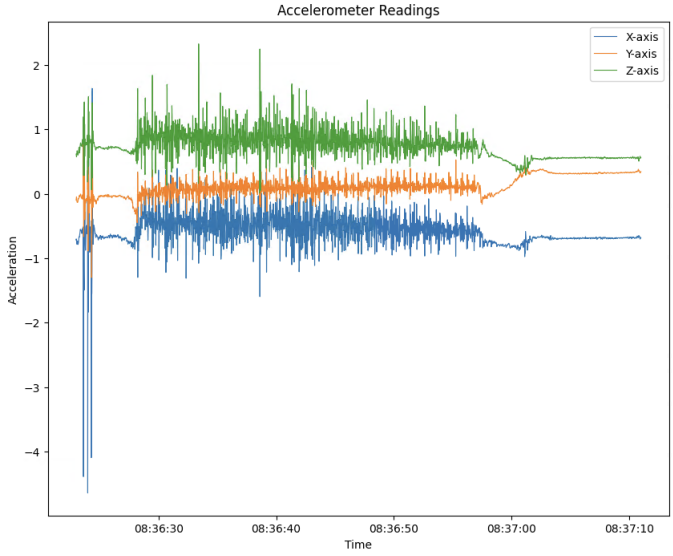
Task 3 involves holding a steady posture, including lifting, holding the posture and drop movements. The variance over time on the filtered gyro_z signal provides the cleanest sepeartion between lift, hold, and drop movements. Figure 4.10d illustrates how the start and end times of Task 3 are detected using variance analysis. The filtered gyro_z data (orange) is plotted alongside the window variance (green), which captures significant motion variations. The variance envelope captures the periods of movement and stability, which helps us identify the fluctuations during the task window. The green marker represents the start time when the variance dropped below the defined threshold (0.1) and remained stable for 10 seconds, indicating the moment the patient held their arms steady. The red marker represents the end time, when the variance rose again indicating the moment the patient lowered their hands.

4.2.4 Task 4 - Hand open and closing

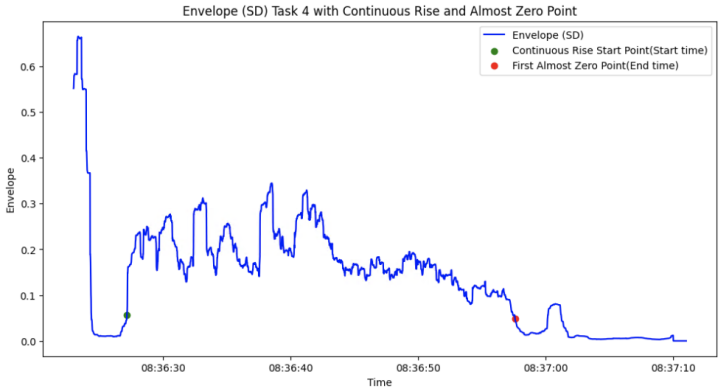
Task 4 involves repeatedly opening and closing the hand as quickly as possible while keeping the arm extended forward. The following plots illustrate the accelerometer and gyroscope readings recorded during this task, highlighting the motion patterns and the start/end detection process.

Accelerometer Readings

Figure 4.11a illustrates the acceleration data across the X, Y, and Z axes during Task 4. As mentioned earlier, the initial three distinct spikes indicate the patient tapping



(a) Plot showing the acceleration of X, Y, and Z-axes from the accelerometer data for Task 4 within the given time frame.



(b) Plot showing the envelope of the accelerometer data, start (green marker) and end (red marker) times of Task 4.

Figure 4.11: Task 4: Acceleration Data and Start & End Time markers

the accelerometer to mark the beginning of the task. In the subsequent readings, we notice that the X and Y axes record continuous fluctuations throughout the task, which indicates repeated hand movements. The Z-axis shows small oscillations likely due to the arm movement as the patient opens and closes their hand. We also notice that towards the end of the task, the fluctuations gradually decrease, indicating a reduction in the movement.

Task 4 Detection Using Standard Deviation Envelope

Task 4 involves hand opening and closing which is a consistent and repetitive motion. The standard deviation envelope on acceleration magnitude effectively captures this motion and help us determine the start and end times. Figure 4.11b visualizes the envelope generated using the standard deviation of the accelerometer magnitude, helping to detect the start and end times of Task 4. The blue curve represents the standard deviation envelope, which illustrates the movement intensity over time. The green marker indicates the start time of the task, which is detected when there is a continuous rise in the envelope that exceeds the predefined threshold (0.05). The red marker identifies the end time of the task, which is identified as the first point where the envelope value falls below the predefined "almost zero" threshold(0.01).

4.2.5 Task 5 - Arm Rotation

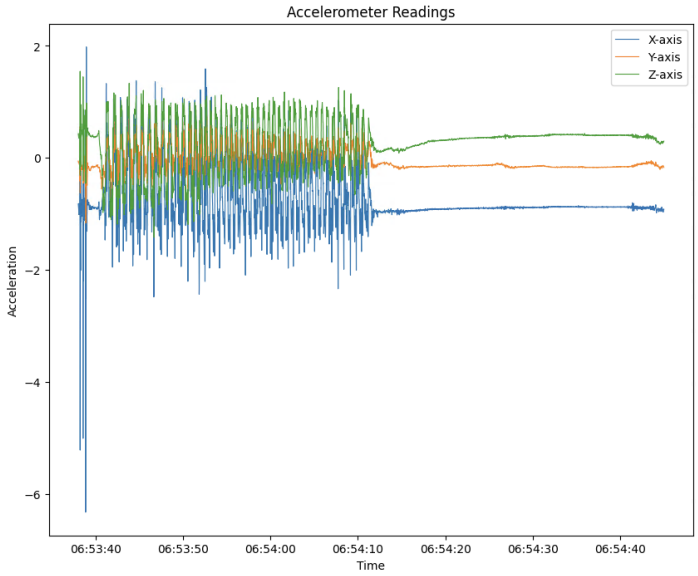
Task 5 involves alternately rotating the palm towards the the ceiling and floor as quickly as possible while keeping the arm extended forward. The following plots illustrate the accelerometer and gyroscope readings recorded during this task, highlighting the motion patterns and the start/end detection process.

Accelerometer Readings

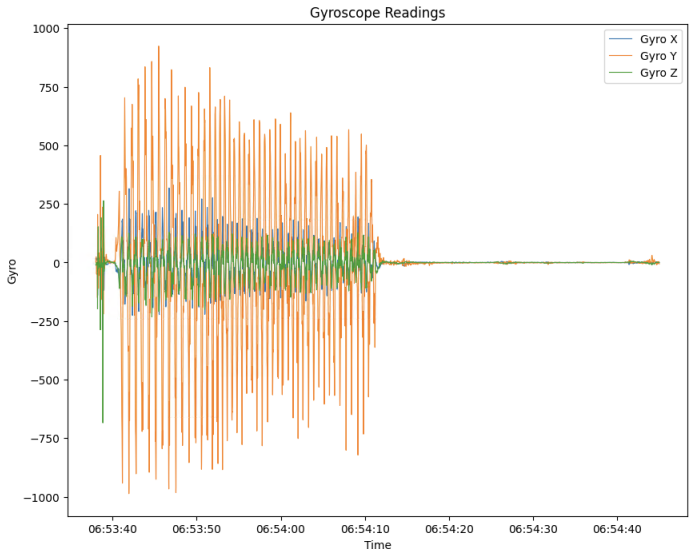
Figure 4.12a shows the acceleration data across the X, Y, and Z axes during Task 5. The initial three distinct spikes correspond to the patient tapping three times on the accelerometer to indicate that they have begun the task. We notice the high-frequency oscillations in the X and Y axes, which represent the rapid rotational movement. The fluctuations in the Z-axis are likely due to the forearm adjustments while rotating the arm.

Gyroscope Readings

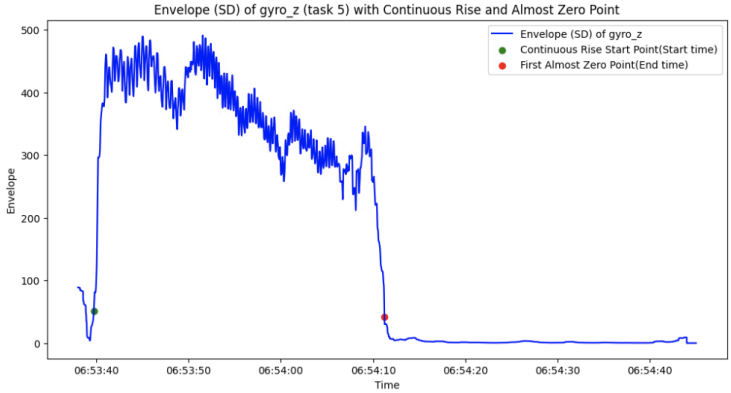
Figure 4.12b visualizes the angular velocity across the X, Y, and Z axes during Task 5. The significant gyroscope fluctuations, especially in the Y-axis, represent the rapid arm motion. We notice a pattern of alternating peaks and troughs in the Y-axis, which confirms the repeated inward and outward arm rotation. The minor fluctuations in the X and Z axes suggest small, unintentional arm or elbow adjustments.



(a) Plot showing the acceleration of X, Y, and Z-axes from the accelerometer data for Task 5 within the given time frame.



(b) Plot showing the gyroscope data of X, Y, and Z-axes for Task 5 within the given time frame.



(c) Plot showing the filtered gyroscope of the Z-axis, envelope generated using standard deviation, start (green marker), and end (red marker) times for Task 5.

Figure 4.12: Task 5: Acceleration, Gyroscope Data and Start & End Time markers

Task 5 Detection Using Standard Deviation Envelope

Task 5 involves primarily rotating motions, because of this we chose to focus our analysis on Gyroscope signals. The standard deviation envelope on gyro_z signal helps us capture the rotational intensity to determine the start and end times. Figure 4.12c illustrates the envelope generated using the standard deviation of the gyroscope Z-axis which is used to detect the start and end times of Task 5. The blue curve represents the standard deviation envelope which illustrates the rotational movement intensity over time. The green marker marks the start of the task, which is detected when the envelope rises above the predefined threshold (50). The red marker marks the end of the task which is detected when the envelope value drops below the predefined "almost zero" threshold.

4.2.6 Task 6 - Foot Stamping

Task 6 involves raising the heel and stomping it on the ground repeatedly while holding the knee with the same-side hand. The following plots illustrate the accelerometer and gyroscope readings recorded during this task highlighting the motion patterns, filtering effects, and the start/end detection process.

Accelerometer Readings

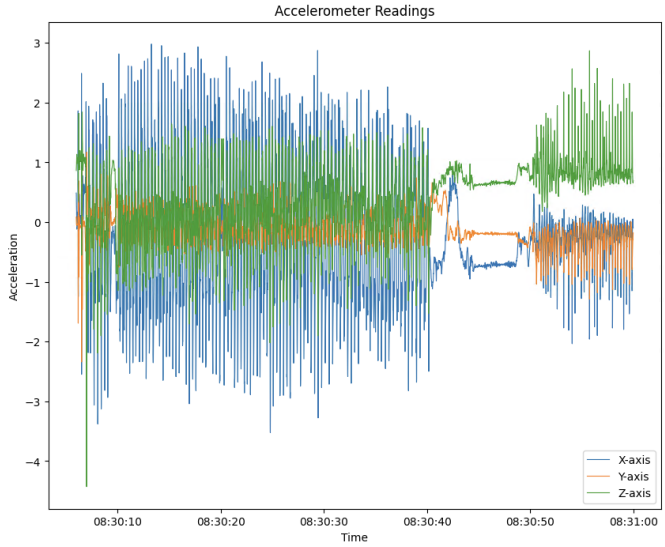
Figure 4.13a shows the acceleration data across the X, Y, and Z axes during the task. In this instance, the initial three taps, which are usually used to indicate the start of the next task, are immediately followed by the actual movement. This results in no visible gap between the taps and the beginning of the task itself. The Z-axis (green line) represents the up-and-down heel motion during heel stamping. The X and Y axes show periodic fluctuations which represent the hand movements while holding the knee and any minor posture adjustments.

Gyroscope Readings

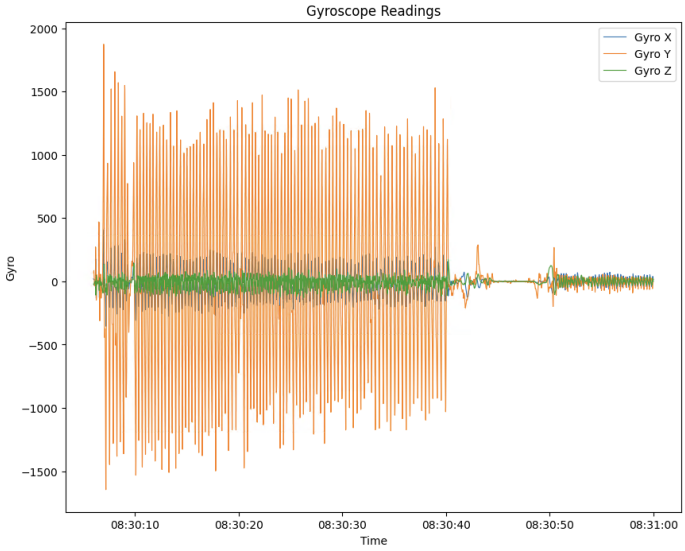
Figure 4.13b shows the angular velocity across the X, Y, and Z axes during Task 6. As mentioned earlier, there is very minimal delay between the three taps, and the patient performing the task of heel stamping. The Z-axis gyroscope shows consistent fluctuations aligning with the heel stamping motion. The Y-axis also exhibits strong fluctuations because while the patient is holding their knee, the wrist tilts forward and backward. We choose gyrpo_z for envelope analysis because it provides a more stable and reliable representation of the foot stamping.

Task 6 Detection Using Standard Deviation Envelope

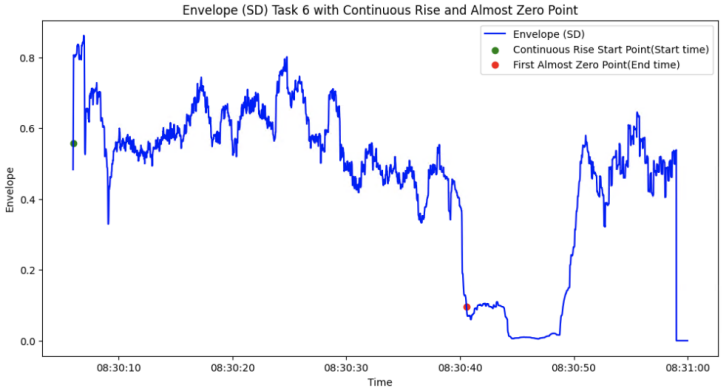
Figure 4.13c illustrates the gyroscope Z-axis envelope generated using the standard deviation analysis to identify the start and end times of Task 6. The green marker signifies the start time. Unlike previous tasks, there is no pause or stabilization period



(a) Plot showing the acceleration of X, Y, and Z-axes from the accelerometer data for Task 6 within the given time frame.



(b) Plot showing the gyroscope data of X, Y, and Z-axes for Task 6 within the given time frame.



(c) Plot showing the filtered gyroscope of the Z-axis, envelope generated using standard deviation, start (green marker), and end (red marker) times for Task 6.

Figure 4.13: Task 6: Acceleration, Gyroscope Data and Start & End Time markers

after the taps, the patient starts the task immediately after three taps. It could mean that this patient didn't need much time to get into position and begin the task, unlike the tasks that may require fine motor control. However, this may not be the case for other patients as they might take longer to transition into the movement. The red marker (end time) is determined when the movement intensity drops below the predefined "almost zero" threshold (0.1).

In this instance, we also need to note that the identified start time slightly overlaps with the identifier taps. This appears to be patient-specific rather than a systemic issue. Our current methodology accounts for the potential delays before the task movement as we have seen with other tasks, however, in the future we could explore ways to specifically distinguish the identifier taps from the stamping movement to improve the segmentation accuracy.

4.2.7 Task 7 - Stand up and Walking

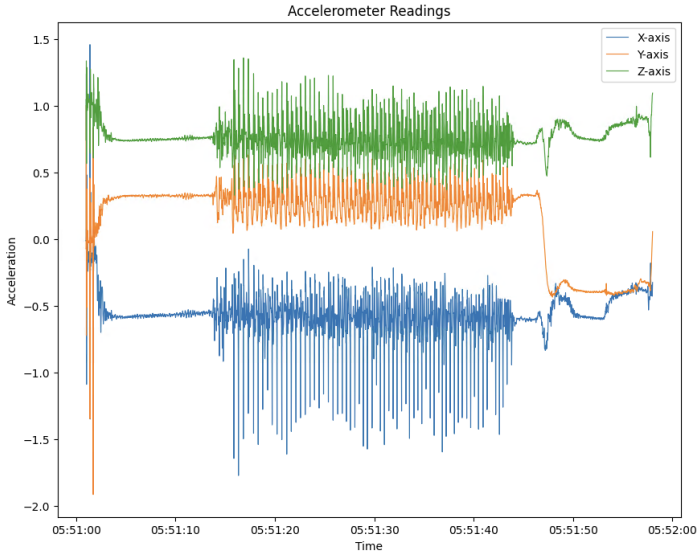
Task 7 involves a series of movements: standing up, walking in a straight line, turning, and repeating the movement until the timer goes off. This task involves both static and dynamic phases. The goal is to detect when the participant starts walking after standing up and when they stop walking. The following plots illustrate the accelerometer readings recorded during this task highlighting the motion patterns and the start/end detection process.

Accelerometer Readings

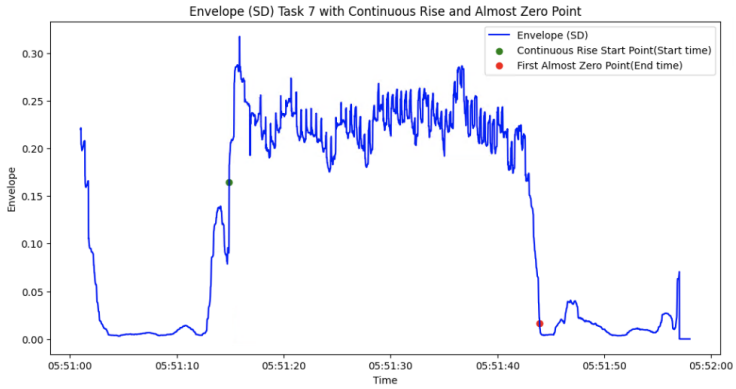
Figure 4.14a illustrates the acceleration data across the X, Y, and Z axes during the task. The three distinct spikes at the beginning of the window are the identifier taps. The acceleration spike we notice after the stabilization period corresponds to the patient pushing themselves off the chair. This is seen in the Z-axis (green line), which represents the vertical movement. After standing, we notice a brief pause followed by repetitive fluctuations that represent the walking cycles. The X and Y axes capture the lateral and forward movement of the body while walking and the Z-axis captures the up-and-down motion of walking.

Task 7 Detection Using Standard Deviation Envelope

Figure 4.14b shows the standard deviation envelope of acceleration data used to determine the start and end times of the task. The algorithm detects the continuous rise in movement and marks the start time when the patient begins walking after standing up. This is illustrated with a green marker. The algorithm ensures that the standing-up phase is not considered part of the walking task. When the patient has stopped walking, when the envelope drops below the predefined threshold (0.1) we identify the end time illustrated with a red marker.

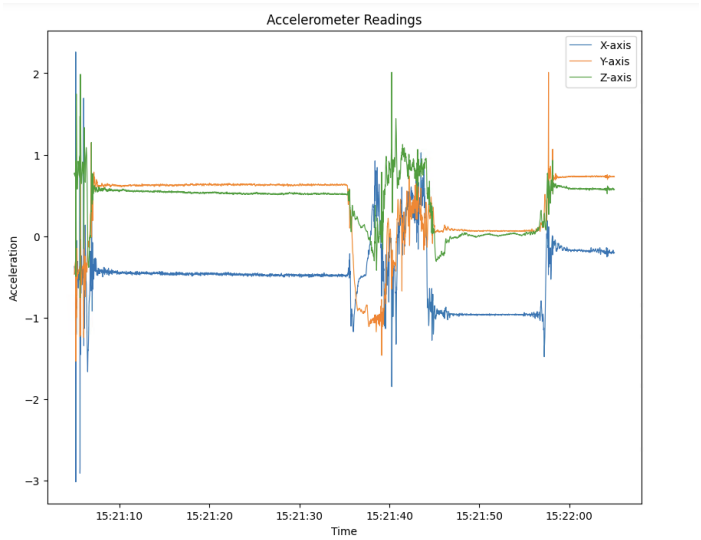


(a) Plot showing the acceleration of X, Y, and Z-axes from the accelerometer data for Task 7 within the given time frame.

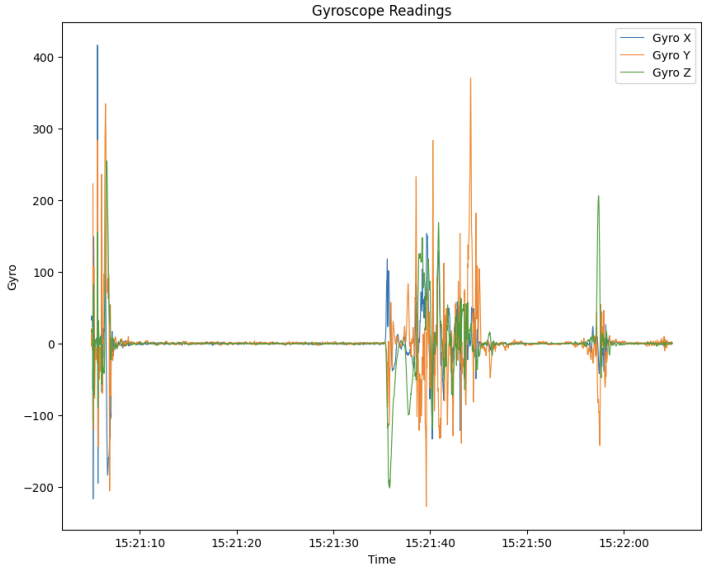


(b) Plot showing the envelope of the accelerometer data, start (green marker) and end (red marker) times of Task 7

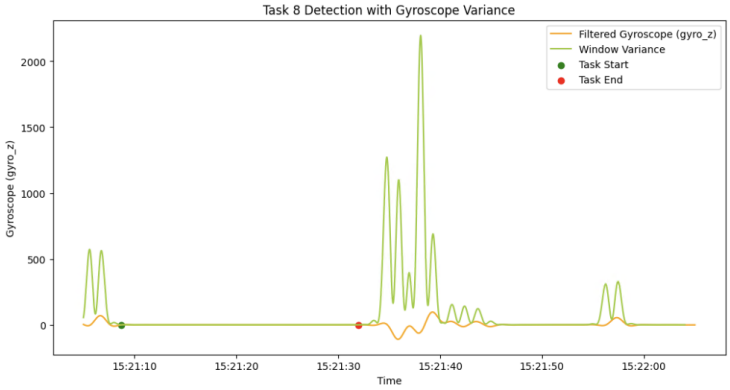
Figure 4.14: Task 7: Acceleration and Start & End Time Markers



(a) Plot showing the acceleration of X, Y, and Z-axes from the accelerometer data for Task 8 within the given time frame.



(b) Plot showing the gyroscope data of X, Y, and Z-axes for Task 8 within the given time frame.



(c) Plot showing the filtered gyroscope of Z-axis, envelope generated using window variance, start (green marker) and end (red marker) for Task 8.

Figure 4.15: Task 8: Acceleration, Gyroscope Data and Start & End Time Markers

4.2.8 Task 8 - Standing still

Task 8 requires the patient to stand up straight with arms crossed and remain still. The following plots illustrate the accelerometer and gyroscope readings recorded during this task, highlighting the motion patterns, filtering effects, and the start/end detection process.

Acceleromter Readings

Figure 4.15a illustrates the acceleration data of the X, Y, and Z axes during Task 8. The three distinct spikes correspond to the indicator taps that the patient taps their wrist to signal the start of the task. We can see that once the patient stands, the acceleration stabilizes. The acceleration values along the X, Y, and Z axes remain relatively steady, indicating that the patient has remained still. We notice a brief increase in acceleration after the end of the task, likely due to the patient relaxing.

Gyroscope Readings

Figure 4.15b shows the angular velocity along the X, Y, and Z axes during Task 8. The three brief spikes are the indicator taps by the patient. After this, we notice that the gyroscope data remains close to zero confirming minimal movement.

Task 8 Detection Using Variance-Based Envelope

Figure 4.15c illustrates the filtered gyroscope of the Z-axis (gyro_z) with a window variance envelope which is used to identify the start and end times of the task. After the identifier taps, the variance of the gyroscope remains close to zero. The green marker signifies the start time. The red marker is the end time when there is a rise in variance which indicates that the patient has resumed movement.

4.3 Validation and Evaluation

The Activity Task Marker Module identifies the start time of the tasks using the acceleration and gyroscope data from the accelerometer. To assess the accuracy of the algorithm, we validate the detected start times against two reference datasets.

1. **Diary data:**
    - This dataset consists of data from 251 patients, 15 tasks each.
    - It contains self-reported task start times that patients record before performing each task.
2. **Expert-Annotated Data:**
    - This dataset consists of data from 15 patients, 15 tasks each.

- It contains the task start times that are manually labeled by the researchers based on visual inspection of the signals.
- This dataset serves as our benchmark for evaluating the detection accuracy.

4.3.1 Understanding the Results

Table 4.3 summarizes the average time difference and standard deviation for both validation datasets.

Validation Method	Average Time Difference (seconds)	Standard Deviation (seconds)
Diary Data (251 participants)	24.68	68.32
Expert-Annotated Data (15 participants)	32.73	63.22

Table 4.3: Validation Results: Average Time Difference and Standard Deviation

Comparison against Diary Dataset

Since the patients write down their start times for a task followed by three identifier taps before they begin the movement, a slight delay in the actual start times is expected. The average delay of 24.68 seconds aligns with the time taken by the patient to prepare for the task after noting it down. The higher standard deviation of 68.32 seconds may suggest that some patients have begun performing the tasks immediately and while some have taken longer time.

Validation against Expert-Annotated Dataset

Expert labeled times are a close representation of the actual start times. The researchers label the times manually after observing the accelerometer signals manually. The average delay of 32.73 seconds is slightly higher compared to the diary dataset but still within the expected range. The standard deviation is 63.22 seconds which is lower than the diary dataset, which suggests more consistency in movement based detected verses the self-reported times.

If the algorithm does not detect the task start times, it skips that task and moves on to detecting the start times for the next task. The results in the Table 4.3 only include the tasks where a task time was successfully detected. The undetected tasks do not influence the average time difference or standard deviation. This means that the reported values reflect only the detected tasks rather than an overall detection rate.

4.3.2 Understanding the 30 seconds delay

The patients perform eight tasks systematically in a session. Each task is separated by three identifier taps before beginning the actual task movement. Typically, each

task lasts 20-30 seconds. The observed 30-second delay in detected start times may result from both patient behavior and signal processing constraints.

There are a few possible explanations for this delay. We have observed that some patients have taken a brief pause before starting the next task, while others begin the next task immediately. These pauses could contribute to the time difference variations we observe. The three taps serve as the indicator between tasks, but if some taps are less distinct the algorithm may detect the start times a little later than expected. The envelope-based detection model smooths the movement signal to reduce the noise and improve reliability while detecting the start time. However, this smoothing process introduces a small delay, as the algorithm checks for the stable movements over noise. It causes the algorithm to detect the task start time a little later than the expert-annotated times. Essentially, the algorithm waits a bit longer to confirm the movement onset. While this helps us prevent false positive detections, it also pushes the start time a little further than it should be. We have fine-tuned the smoothing parameters and tested various tap detection thresholds for each task to find the balance between accuracy and reliability. The current settings minimize the false detections while still ensuring that the start times are detected consistently. It is also worth mentioning that our validation set is quite small (15 participants, 15 tasks each). The observed delay might not fully provide us insight into how the algorithm performs on a larger sample. The variability in time differences, as reflected by the standard deviation, is also partially influenced by individual behavior of the patients and not only just a systematic delay caused by the algorithm.

The algorithm consisting of Tap Detection Module and Activity Task Marker Module successfully detects the task start and end times. It provides a consistent and reliable way to segment the tasks. While there is some variation in the start times, it is expected, given the differences in how patients perform the tasks and how the signal is processed. Overall, the detected start times align well with the real-world movement patterns and patient behavior, which demonstrates the algorithm's effectiveness in task segmentation.

Chapter 5

Discussion and Conclusion

5.1 Contributions to the field

This research contributes to automating movement analysis in Parkisnon's disease monitoring by developing a new and reliable algorithm to detect the pre-activity taps, start and end times of the tasks using accelerometer data. Currently, the researchers rely on manual annotation, which can be time-consuming and inconsistent due to inter-observer variability. The Tap Detection Module identifies the pre-activity taps and the Activity Task Marker Module provides an automated and reliable way to detect start and end times by detecting the key movements when performing the tasks. The algorithm is consistent, scalable and objective in segmenting the tasks.

One of the major contributions of this work is the fine-tuned, envelope based detection method. We have carefully tested and adjusted the detection threshold for each task to make the system more scalable and reliable. By refining the detection thresholds for various tasks we were able to create a scalable and robust approach for identifying the movement onset with minimal false detections of the task start times.

From technical perspective, this research contributes by:

- Developing a structured method to detect the key movements using the data from a wrist-worn accelerometer.
- Optimizing an envelope-based detection method to work across various structured tasks.
- Validating the accuracy of our method by comparing the detected task start times with both patient reported start times and expert-annotated start times.

Beyond the technical impact, this work also has real world applications, particularly in Parkinson's disease monitoring and treatment evaluation:

- It helps in tracking the Parkinson's symptoms progression over time. By monitoring the performance on the eight structures tasks, the doctors can assess whether a patient's motor abilities are improving, stabilizing or worsening.

- It helps in evaluating the effects of medication. The study includes OFF and ON tests (before and after medication intake), by comparing how they perform during these tests, the doctors can assess how well a patient responds to treatment.

- It reduces the manual effort taken for movement tracking making it more practical for long term monitoring

- It has potential to scale for broader applications, like rehabilitation programs, sports science and wearable health monitoring.

By testing our algorithm on the larger dataset of 251 participants, the study shows that the automated task segmentation can work on a wide scale. It can be a valuable tool for Parkinson's research.

5.2 Limitations and Challenges

We have faced several technical and methodological challenges while developing an automated task segmentation system for Parkinson's disease monitoring. While the final algorithm is a reliable and scalable, there were certain limitations that had to be addressed during the development process.

5.2.1 Reading and Processing Accelerometer Data

One of the earliest challenges was to correctly read the Axivity AX6 accelerometer data in Python. The CWA file format of the accelerometer data involved specialized treatment of the files and many existing Python packages either failed to fully process the data or there was data loss. While Axivity provides better support for R language, integrating the 'CwaData' package in Python took considerable effort to ensure that there is no signal distortion or missing values. This was important as any loss or misinterpretation of the data would impact all the later analysis. The project was developed in Python to enable for future integration with machine learning pipelines.

5.2.2 Task-Specific Detection and Threshold Optimization

There was no one-size-fits-all approach while developing the method to detect the start and end times of the tasks. Each of the eight standardized tasks involved different movement patterns and intensity, which is why we required a task-specific approach to determine the thresholds and the method. For example:

- Tasks like "Finger tapping" and "Hand Open/Closing" needed a higher sensitivity to short-duration motion changes because of the repeated motions, which standard deviation envelopes handled better.

- Tasks like "Resting while sitting" and "Standing Still" involved little to no movement, which required a more sensitive variance-based envelope approach to detect the subtle changes.

- Tasks like "Arm Rotation" and "Lifting Arms" which involved rotational movement required gyroscope data (gyro_z) rather than accelerometer data, a different feature extraction method.

For each task, we had done extensive trial and error to determine the best threshold settings that balanced the accuracy and robustness. This was labor-intensive and had to be re-evaluated at multiple stages.

5.2.3 Low-Pass Filtering and Signal Smoothing

We used Butterworth low-pass filter to remove the noise and smooth the accelerometer data. While this has helped in removing the unwanted noise effectively, it had introduced a slight delay in detecting the movement onset. This was particularly noticeable in the tasks where the patients hadn't performed the indicator taps properly. This has effected our accuracy slightly, which can be seen when comparing the algorithm detected start times with the expect-annotated start times. We observe the discrepancy of 30 seconds on average. The system was designed to prioritize accuracy and be reliable, avoiding any false positives but this smoothing effect remains a minor limitation.

5.2.4 Variability in Patient Behavior

There was some inconsistency in how the patients performed the tasks as they were performed at home without supervision. We observe that some people have taken brief breaks between the tasks, while other completed alt the eight tasks in one continuous sequence. This added some complexity to task segmentation as the algorithm had to handle both back-to-back transitions and irregular gaps. Additionally, the diary recorded times are prone to human error. There were also some inaccuracies in time notation which required some pre-processing and standardization.

Despite these challenges, careful algorithm design, fine-tuning and iterative testing helped us design a system that can achieve reliable performance across all tasks. While the small delays in detection and occasional missing tasks remain areas for improvement, the current algorithm provides a robust foundation for future refinements.

5.3 Future Work

This research lays the foundation for identifying the Activity Tests in Parkinson's Disease Patients using accelerometer data but there is some room for development. Now that we have the identified start and end times of the tasks, we could train

deep learning models to classify the tasks. This could be achieved by analyzing the movement patterns within the identified time-frame of each task's accelerometer data. If successful, this could significantly reduce or even eliminate the need for manual annotations.

With some adjustments, this research could be applied beyond monitoring Parkinson's Disease patients. It could be applied to other fields such as rehabilitation therapy and sports science. The algorithm could be used to track the rehabilitation process, analyze the movement patterns and gain deeper insights into one's progress. As wearable technology advances, improving the task detection and segmentation algorithms using accelerometer data may result in more accurate movement pattern analysis. This could benefit both clinical research and real-world applications.

5.4 Conclusion

In summary, this research presents an approach that is practical and validated for automating Activity tests identification in Parkinson's patients' movement analysis using Accelerometer data. By combining tap detecting and envelope-based task identification using signal processing methods, we developed a consistent and scalable solution for real-world deployment. While some variability exists because of patient specific factors, the algorithm can serve as a reliable alternative to manual annotation and for long-term monitoring.

References

Auluck, P., & Bonini, N. (2002). Pharmacological prevention of parkinson disease in drosophila. *Nature Medicine*, *8*, 1185–1186. doi: 10.1038/nm1102-1185

Axivity Ltd. (n.d.). *Ax6.* Retrieved from `https://axivity.com/product/ax6` (Online; accessed January 10, 2024)

Bickley, L., & Szilagyi, P. G. (2012). *Bates' guide to physical examination and history-taking.* Lippincott Williams & Wilkins.

Cespedes, I., Insana, M., & Ophir, J. (1995). Theoretical bounds on strain estimation in elastography. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, *42*(5), 969-972.

Johnson Jr, C. R., Sethares, W. A., & Klein, A. G. (2011). *Software receiver design: Build your own digital communication system in five easy steps.* Cambridge University Press.

Kandel, E. R., Schwartz, J. H., Jessell, T. M., Siegelbaum, S., Hudspeth, A. J., & Mack, S. (2000). Principles of neural science. In *Principles of neural science* (Vol. 4, p. 1227-1246). New York: McGraw-Hill.

Keijsers, N. L., Horstink, M. W., & Gielen, S. C. (2006). Ambulatory motor assessment in parkinson's disease. *Movement Disorders: Official Journal of the Movement Disorder Society*, *21*(1), 34–44. Retrieved from `https://doi.org/10.1002/mds.20633` doi: 10.1002/mds.20633

LeMoyne, R. (2013). Wearable and wireless accelerometer systems for monitoring parkinson's disease patients—a perspective review. *Advances in Parkinson's Disease*.

Nygårds, M. E., & Sörnmo, L. (1983). Delineation of the QRS complex using the envelope of the ECG. *Medical and Biological Engineering and Computing*, *21*, 538-547.

Robertson, D. G. E., & Dowling, J. J. (2003). Design and responses of butterworth and critically damped digital filters. *Journal of Electromyography and Kinesiology*, *13*(6), 569-573.

Samii, A., Nutt, J. G., & Ransom, B. R. (2004). Parkinson's disease. *The Lancet*, *363*(9423), 1783-1793.

Schweiger, D., Critcher, S., Freeborn, T. J., & Hibberd, E. (2020). Throwing event detection using acceleration magnitude collected with wrist-worn sensors. In *2020 southeastcon* (p. 1-7). doi: 10.1109/SoutheastCon44009.2020.9249695

SciPy Community. (2008–2023). SciPy: Open source scientific tools for python [Computer software manual]. Retrieved from `https://www.scipy.org/`

SciPy Community — scipy.signal.butter. (2008–2023). scipy.signal.butter [Computer software manual]. Retrieved from `https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.butter.html` (SciPy documentation, [Online; accessed January 10, 2024])

SciPy Community — scipy.signal.filtfilt. (2008–2023). scipy.signal.filtfilt [Computer software manual]. Retrieved from `https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.filtfilt.html` (SciPy documentation, [Online; accessed January 10, 2024])

Silva, M., Exadaktylos, V., Ferrari, S., Guarino, M., Aerts, J. M., & Berckmans, D. (2009). The influence of respiratory disease on the energy envelope dynamics of pig cough sounds. *Computers and Electronics in Agriculture*, *69*(1), 80-85.

Technical Developer Guide, Axivity Ltd. (n.d.). Technical developer guide [Computer software manual]. Retrieved from `https://axivity.com/userguides/wax9/technical/` (Online; accessed January 10, 2024)

Thalmayer, A., Zeising, S., Fischer, G., & Kirchner, J. (2020). A robust and real-time capable envelope-based algorithm for heart sound classification: Validation under different physiological conditions. *Sensors*, *20*, 972. doi: 10.3390/s20040972

Varghese, T., & Ophir, J. (1998). Characterization of elastographic noise using the envelope of echo signals. *Ultrasound in Medicine & Biology*, *24*(4), 543-555.

Volkmann, J., Moro, E., & Pahwa, R. (2006). Basic algorithms for the programming of deep brain stimulation in parkinson's disease. *Movement Disorders: Official Journal of the Movement Disorder Society*, *21*(S14), S284-S289.

Weinstein, E., & Weiss, A. (1984). Fundamental limitations in passive time-delay estimation–part ii: Wide-band systems. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, *32*(5), 1064-1078.

Winter, D. A. (1990). *The biomechanics and motor control of human movement*. Wiley-Interscience, 52.