# A Study of Ensemble Machine Learning Model Architectures for Parkinson's Disease Detection and Freezing of Gait Forecasting

by

Tahjid Ashfaque Mostafa

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

Department of Computing Science

University of Alberta

# Abstract

Parkinson's Disease (PD) is a major progressive neurological disorder and is extremely difficult to diagnose PD [91] since there are no defined medical tests for this task. The existing approach involves a combination physical examinations, neuroimaging and demographic analysis performed by expert medical professionals. This process is both time and resource draining as well as being prone to human error and bias. Moreover, the motor symptoms might not appear until the advanced stages of the disease, the diagnosis often does not provide ample time to administer preventive measures. Computer Aided Diagnosis (CAD) systems have been gaining popularity in recent years, but these solutions are not without their own shortcomings. In this work, non-invasive approaches to identify PD and monitor the progression of one of the motor symptoms of PD using deep learning based techniques are analyzed.

We explore various approaches to discuss PD case from control using Magnetic Resonance (MR) T1 images of the brain, one of the most popular neuroimaging techniques, non-invasive and generating high resolution images in the soft tissue. We experimented with some Convolutional Neural Network (CNN) models of ImageNet Large Scale Visual Recognition Challenge (ILSVRC) with whole brain, extracted Gray Matter (GM) and White Matter (WM) scans. We also propose multiple ensemble architectures combining the ILSVRC models. The detection accuracy increases drastically when we focus on the extracted GM and WM regions from the MR images instead of using the whole brain scans. ILSVRC Deep Learning (DL) models pretrained on the

ImageNet dataset perform relatively better than when they are trained solely on the MRI scans. The proposed solutions outperform state of the art existing methods on similar datasets.

One of the major obstacles in applying learning algorithms to this task is lack of properly labeled training data. So our finding that training on unrelated data might increase the performance of DL models is a possible solution. We also perform occlusion analysis and determine brain areas are relevant in the DL architectures decision making process. This was to further narrow down the regions of interest. Focusing on the identified relevant regions might be helpful in achieving the same performance while reducing the amount of data needed to be processed.

Freezing of Gait (FOG) is an impairment that affects the majority of patients in the advanced stages of PD, defined as a short period of time when the patient fails to move forward, despite attempting to do so. The patients describe this event as a sudden feeling of their feet being stuck to the ground. FOG can lead to sudden falls and injuries, negatively impacting the quality of life for the patients and their families. Rhythmic Auditory Stimulation (RAS) can be used to help patients recover from FOG and resume normal gait. However, even if FOG is detected in early stages, RAS might not be effective due to the latency between the start of a FOG event, detection and initialization of RAS. In the second section, I propose a system capable of both FOG prediction and detection using signals from tri-axial accelerometer sensors. This approach will be useful in initializing RAS with minimal latency. I compared the performance of several time frequency analysis techniques, including moving windows extracted from the signals, handcrafted features, Recurrence Plots (RP), Short Time Fourier Transform (STFT), Discreet Wavelet Transform (DWT) and Pseudo Wigner Ville Distribution (PWVD) with DL based Long Short Term Memory (LSTM) and CNN. I also propose three Ensemble

Network Architectures that combine all the time frequency representations and DL architectures. Experimental results show that our ensemble architectures significantly improve the performance compared with existing techniques on benchmark dataset. Our research group also collaborated with A. T. Still University to collect motion data for a group of PD patients, some of whom experienced FOG during the data collection. I also applied the methods proposed in the second section on this data to identify the instances of FOG.

# Preface

The contents of this thesis have been published or are under review in multiple journals and conferences. Parts of the contents of both Chapter 2 and Chapter 3 contain our work on Parkinson's Disease (PD) detection from MRI images of the brain and has been published in IEEE Bioinformetics and Bioengineering (BIBE) 2020. Contents from Chapter 2 and Chapter 3 have also been submitted for publication to International Symposium on Visual Computing (ISVC) 2021. Parts of contents of Chapter 2 and entire Chapter 4 detail our work on Freezing Of Gait (FOG) Progression Monitoring for PD using Time Frequency Representation Techniques and Neural Network Architectures. This work has been submitted for publication in "Perception and Intelligence Driven Sensing to Monitor Personal Health" Special Edition of the Sensors Journal.

**Publications during MSc Study**

- **Tahjid Ashfaque Mostafa** and Irene Cheng, "Parkinson's disease detection using ensemble architecture from MR images*," 2020 IEEE 20th International Conference on Bioinformatics and Bioengineering (BIBE), 2020. doi:10.1109/bibe50027.2020.00167.

- **Tahjid Ashfaque Mostafa** and Irene Cheng, Parkinson's disease detection with ensemble architectures based on ILSVRC models, 2020. arXiv: 2007.12496[eess.IV].

- **Tahjid Ashfaque Mostafa**, Sara Soltaninejad, Tara L. McIsaacand and Irene Cheng, A Comparative Study of Time Frequency Representation Techniques for Freeze of Gait Detection and Prediction. Preprints 2021, 2021080347 (doi: 10.20944/preprints202108.0347.v1).

# Acknowledgements

Completing my MSc has been a life altering experience and I believe it wouldn't have been possible without the guidance and support I have received throughout my program from many great people.

First of all, I would like to thank my supervisor Dr. Irene Cheng, whose knowledge, expertise and guidance were vital in my research. The patience she showed in guiding me and my work as well as the support provided by her were invaluable.

I acknowledge the funding provided by NSERC which helped me in continuing my research. I appreciate the assistance and guidance of Dr. Sara Soltaninejad throughout my program.

I would also like to thank my thesis examiner committee, Dr. Anup Basu and Dr. Russ Greiner for their indispensable guidance and for providing me with the opportunity to successfully complete my thesis.

Finally, I would like to express my gratitude to every person who helped me in my journey directly and indirectly in these trying times.

# Contents

**Acronyms** **xii**

**1 Introduction** **1**
1.1 Parkinson's Disease - Importance, Causes and Symptoms . . . 2
1.2 Diagnosis of Parkinson's and Challenges . . . 3
1.3 Computer Aided Diagnosis of Parkinson's Disease . . . 4
1.4 Freezing of Gait (FOG) - Major Symptom of Parkinson's . . . 5
1.5 Application of Deep Learning for Parkinson's Analysis and Challenges . . . 7
1.6 Contributions . . . 9
1.7 Organization of the Thesis . . . 10

**2 Background Material** **11**
2.1 Background and Literature Review for Parkinson's Disease detection . . . 11
2.2 Background and Literature Review for Freezing of Gait Detection and Prediction . . . 14
2.3 Performance Evaluation Criteria . . . 18
2.3.1 Detection Accuracy . . . 19
2.3.1.1 Precision, Recall/Sensitivity, Specificity, $F_\beta$ Score 19
2.3.2 Matthews Correlation Coefficient . . . 20

**3 Parkinson's Disease Detection from Magnetic Resonance Imaging** **21**
3.1 Introduction . . . 21
3.2 Data . . . 22
3.3 Preprocessing . . . 24
3.4 Model Structure . . . 30
3.4.1 Ensemble Architecture - Model 1 . . . 30
3.4.2 Ensemble Architecture - Model 2 . . . 31
3.4.3 Ensemble Architecture - Model 3 . . . 32
3.4.3.1 Sub Architecture 1 - TriNet1 . . . 33
3.4.3.2 Sub Architecture 2 - QuadNet . . . 34
3.4.3.3 Sub Architecture 3 - TriNet2 . . . 35
3.5 Experimental Results . . . 35
3.6 Discussion . . . 52
3.7 Occlusion Analysis . . . 57
3.7.1 Occlusion Analysis for Modified ResNet . . . 57
3.7.2 Occlusion Analysis for Ensemble Architecture - Model 1 61
3.7.3 Occlusion Analysis for Ensemble Architecture - Model 2 64
3.8 Conclusion . . . 67

**4 Freezing Of Gait Detection and Prediction** **68**
4.1 Introduction . . . 68
4.2 Data . . . 69
4.3 Proposed Method . . . 72
4.3.1 Preprocessing . . . 72
4.3.1.1 Removing Unrelated Data . . . 73
4.3.1.2 Calculating Magnitude of Acceleration for all three axis . . . 73
4.3.2 Data Augmentation . . . 73
4.3.2.1 Signal Segmentation . . . 74
4.3.2.2 Labeling PreFOG class . . . 74
4.3.2.3 SMOTE Oversampling . . . 75
4.3.3 Feature Extraction . . . 76
4.3.3.1 Time and Frequency Domain Features . . . 76
4.3.3.2 Recurrence Plots (RP) . . . 79
4.3.3.3 Short Time Fourier Transform (STFT) . . . 81
4.3.3.4 Discrete Wavelet Transform (DWT) . . . 82
4.3.3.5 Pseudo Wigner Ville Distribution (PWVD) . . . 84
4.4 Model Structure . . . 86
4.4.1 Basic Convolutional Neural Network Architecture (CNN) 86
4.4.2 Basic Bidirectional Long Short Term Memory Architecture (LSTM) . . . 87
4.4.3 Ensemble Architectures . . . 88
4.4.3.1 Stacked Ensemble Model - M7: . . . 89
4.4.3.2 Average Ensemble Model - M8: . . . 90
4.4.3.3 Majority Voting - M9: . . . 90
4.5 Results . . . 90
4.5.1 K-Fold Cross Validation . . . 90
4.5.2 Normalization . . . 91
4.5.3 Experimental Setup . . . 91
4.5.4 Metric Scores . . . 92
4.6 Discussion . . . 102
4.7 Application of Trained Model on Data collected from APDM™ sensors . . . 106
4.7.1 Data collection and Processing . . . 106
4.7.1.1 Sensor types and locations . . . 107
4.7.2 Workflow and Challenges . . . 109
4.8 Conclusions . . . 114

**5 Conclusion And Future Directions** **115**

**References** **117**

# List of Tables


3.1 Demographic Data . . . . . 22
3.2 Results of some related works on similar dataset . . . . . 36
3.3 Results for Resnet with Learning Rate of 0.0001 . . . . . 37
3.4 Results for VGG with Learning Rate of 0.0001 . . . . . 38
3.5 Results for DenseNet with Learning Rate of 0.0001 . . . . . 39
3.6 Results for MobileNet with Learning Rate of 0.0001 . . . . . 40
3.7 Results for ShuffleNet with Learning Rate of 0.0001 . . . . . 41
3.8 Results for SqueezeNet with Learning Rate of 0.0001 . . . . . 42
3.9 Results for Ensemble Model 1 for Whole Brain Scans . . . . . 43
3.10 Results for Ensemble Model 2 for Gray Matter and White Matter 44
3.11 Results for Ensemble Model 2 for Smoothed Gray Matter and White Matter . . . . . 45
3.12 Results for Ensemble Model 3 with Trinet1 for Gray Matter and White Matter . . . . . 46
3.13 Results for Ensemble Model 3 with Trinet1 for smoothed Gray Matter and White Matter . . . . . 47
3.14 Results for Ensemble Model 3 with QuadNet for Gray Matter and White Matter . . . . . 48
3.15 Results for Ensemble Model 3 with QuadNet for Smoothed Gray Matter and White Matter . . . . . 49
3.16 Results for Ensemble Model 3 with TriNet2 for Gray Matter and White Matter . . . . . 50
3.17 Results for Ensemble Model 3 with TriNet2 for smoothed Gray Matter and White Matter . . . . . 51
3.18 Results for Ensemble Model 1 with Pre trained constituent models for Gray Matter and White Matter . . . . . 52

4.1 $F_i$ Features extracted for each $\alpha_i \in \alpha_C$ . . . . . 77
4.2 Results of some existing methods . . . . . 92
4.3 Results of Basic CNN architecture M1 with Recurrence Plot (RP) 93
4.4 Results of Basic CNN architecture M2 with Short Time Fourier Transform (STFT) . . . . . 94
4.5 Results of Basic CNN architecture M3 with Discrete Wavelet Transform (DWT) . . . . . 95
4.6 Results of Basic CNN architecture M4 with Pseudo Wigner Ville Distribution (PWVD) . . . . . 96
4.7 Results of Bidirectional LSTM architecture M5 with Raw Signals 97
4.8 Results of Bidirectional LSTM architecture M6 with Extracted features . . . . . 98
4.9 Results of Stacked Ensemble M7 with Extracted features . . . 99
4.10 Results of Average Ensemble M8 with Extracted features . . . 100
4.11 Results of Majority Voting M9 with Extracted features . . . . 101

# List of Figures


2.1 Sample Images from ImageNet [20] dataset and their position in the WordNet [21], [139] Hierarchy . . . . . 13

3.1 Brain aging with age represented as trajectories. The thick brown line represents healthy aging. The purple line represents rapid brain aging due to injuries and blue line represents brain aging due to generic or environmental issues. The yellow line represents more prominent but stable brain aging due to injuries. X axis represents age and Y axis represents probability. Figure taken from [19] . . . . . 24
3.2 Preprocessing Pipeline . . . . . 25
3.3 Sample MRI scan comparison for a Healthy Control subject and Parkinson's Patient . . . . . 27
3.4 Extracted White Matter (WM) from MRI scans from Figure 3.3 28
3.5 Extracted Gray Matter (GM) from MRI scans from Figure 3.3 29
3.6 Ensemble architecture for whole brain scan: Model 1 . . . . . 31
3.7 Ensemble architecture for Extracted Gray and White Matter Scans : Model 2 . . . . . 32
3.8 Core Architecture . . . . . 33
3.9 Architecture 1 : TriNet1 . . . . . 33
3.10 Architecture 2 : QuadNet . . . . . 34
3.11 Architecture 3 : TriNet2 . . . . . 34
3.12 Relevance Heatmaps for Occlusion of Gray Matter and White Matter images using Pretrained ResNet [40] . . . . . 59
3.13 Relevance per brain area for White Matter for Pretrained ResNet [40] 60
3.14 Relevance per brain area for Gray Matter for Pretrained ResNet [40] 60
3.15 Relevance Heatmaps for Occlusion of White Matter and Gray Matter images using Ensemble Architecture 1 with pretrained constituent models . . . . . 62
3.16 Relevance per brain area for White Matter using Ensemble Architecture 1 with pretrained constituent models . . . . . 63
3.17 Relevance per brain area for Gray Matter using Ensemble Architecture 1 with pretrained constituent models . . . . . 63
3.18 Relevance Heatmaps for Occlusion of White Matter and Gray Matter images using Ensemble Architecture 2 with pretrained constituent models . . . . . 65
3.19 Relevance per brain area for White Matter using Ensemble Architecture 2 with pretrained constituent models . . . . . 66
3.20 Relevance per brain area for Gray Matter using Ensemble Architecture 2 with pretrained constituent models . . . . . 66

4.1 Sensor Placement for Data Collection . . . . . 70

4.2 Proposed Preprocessing, Data Augmentation and Feature Extraction workflow . . . . . 72
4.3 Example of combined Accelerometer signal from Ankle, capturing the motor variations in the gait of a Parkinson's patient, containing Normal gait, followed by a window of PreFOG period (Yellow), and then a FOG event (Red). . . . . 75
4.4 Examples of Recurrence Plot generated from Accelerometer signal from Ankle with a Window size of 2 and $\epsilon$s value of 0.5. . 80
4.5 Examples of Short Time Fourier Transform Plot generated from Accelerometer signal from Ankle with a Window size of 2. . . 82
4.6 Examples of Discrete Wavelet Transformation generated from Accelerometer signal from Ankle with a Window size of 2. . . 84
4.7 Examples of Pseudo Wigner Ville Distribution generated from Accelerometer signal from Ankle with a Window size of 2. . . 85
4.8 Proposed basic CNN Architecture with 4 recurring 2D Convolution blocks, followed by 3 Dense layers. . . . . 87
4.9 Proposed basic Bidirectional LSTM Architecture with 4 recurring Bidirectional LSTM block, followed by a Dense Layer. . . 88
4.10 Proposed Ensemble Architectures, (1) M7 concatenates the output all constituent models, followed by a Dense Layer, (2) M8 Averages the outputs of all constituent models and (3) M9 calculates the majority prediction of all models using mode. . . 89
4.11 Sensor placement for data collection . . . . . 107
4.12 Workflow of monitoring FOG using models trained on Daphnet data . . . . . 109
4.13 Comparison of Freezer vs Non Freezer Accelerometer signals . 111
4.14 Comparison of Freezer vs Non Freezer Accelerometer signals, with detected FOG, PreFOG and Non FOG regions highlighted 113

# Acronyms

**A | B | C | D | E | F | G | H | I | K | L | M | N | O | P | R | S | W**

**A**

**Activities of Daily Living (ADL)**

**Activity Monitoring System (U-AMS)**

**Alzheimer's Disease (AD)**

**Area Under Curve (AUC)**

**Artificial Intelligence (AI)**

**B**

**Basal Ganglia (BG)**

**C**

**Central Nervous System (CNS)**

**Cerebrospinal Fluid (CSF)**

**Computational AnatomTtoolbox (CAT12)**

**Computer Vision (CV)**

**Computer Aided Diagnosis (CAD)**

**Convolutional Neural Network (CNN)**

**D**

**Deep Neural Network (DNN)**

**Deep Learning (DL)**

**Diffuse Lewy body (DLB)**

**Discrete Wavelet Transform (DWT)**

**E**

**Equal Error Rate (EER)**

**Essential Tremor (ET)**

**F**

**Fast Fourier Transform (FFT)**

**Freeze Index (FI)**

**Freezing Threshold (FTH)**

**Freezing Band (FB)**

**Freezing of Gait-Questionnaires (FOG-Q)**

**Freezing of Gait (FOG)**

**G**

**Geometric Mean (GM)**

**Gradient Boosting (GB)**

**Gray Matter (GM)**

**H**

**Healthy Control (HC)**

**Hidden Markov Model (HMM)**

**Human Activity Recognition (HAR)**

**I**

**ImageNet Large Scale Visual Recognition Challenge (ILSVRC)**

**Inertial Measurement Unit (IMU)**

**K**

**K-Nearest Neighbour (KNN)**

**L**

**Learning Rate (LR)**

**Leave-One-Subject-Out (LOSO)**

**Long Short Term Memory (LSTM)**

**M**

**Machine Learning (ML)**

**Magnetic Resonance (MR)**

**Magnetic Resonance Imaging (MRI)**

**Matthews Correlation Coefficient (MCC)**

**Mel Frequency Cepstral Coefficients (MFCCs)**

**Montreal Neurological Institute (MNI)**

**Multiple System Atrophy (MSA)**

**N**

**National Health Service (NHS)**
The United Kingdom National Health Service

**Neural Network (NN)**

**New Freezing of Gait Questionnaire (NFOGQ)**

**O**

**One-Class Support Vector Machine (OC-SVM)**

**P**

**Parkinson Progression Markers Initiative (PPMI)**

**Parkinson's Disease (PD)**

**Parkinsonian Variant of Multiple System Atrophy (MSA-P)**

**Power Threshold (PTH)**

**Power Index (PI)**

**Progressive Supranuclear Palsy (PSP)**

**Pseudo Wigner Ville Distribution (PWVD)**

**R**

**Radial Basis Function (RBF)**

**Radial Basis Function Neural Network (RBFNN)**

**Random Forest (RF)**

**Receiver operating characteristic (ROC)**

**Rectified Linear Unit (ReLU)**

**Recurrence Plot (RP)**

**Recurrent Neural Network (RNN)**

**Region Of Interest (ROI)**

**resting-state functional MRI (rsf-MRI)**

**Rhythmic Auditory Stimulation (RAS)**

**S**

**Short Time Fourier Transform (STFT)**

**Single Photon Emission Computed Tomography (SPECT)**

**Speech Quality Assessment (SQA)**

**Statistical Parameter Mapping (SPM12)**

**structural MRI (sMRI)**

**Substantia Nigra (SN)**

**Superior Temporal Gyrus (STG)**

**Support Vector Machine (SVM)**

**Synthetic Minority Oversampling Technique (SMOTE)**

**W**

**Walking Band (WB)**

**White Matter (WM)**

# Chapter 1

# Introduction

Intricate interactions and co-ordinations between muscles, nerves and the Central Nervous System (CNS), consisting of the brain and spinal cord, are responsible for every movement of the human body, from walking, raising an arm to even the smallest twitch of the eye. Pathological changes within the brain can cause damage to these components, leading to a number of nervous system conditions that cause abnormal, excess or paucity of movements, both voluntary and involuntary [120]. These group of neurological conditions are referred to as movement disorders. Depending on the location and type of damage to the nervous system, movement disorders can be of varying types. Some of the more well known types of movement disorders are Hyperkinesia (excess movements), Dyskinesias (unnatural movements), abnormal involuntary movements, Hypokinesia (reduced amplitude of movement), Bradykinesia (slowness of movement) and Akinesia (complete loss of movement or paralysis) [25]. More than 30 different diseases are classified as movement disorders, some of the most prevalent ones being Alzheimer's Disease (AD), Essential Tremor (ET), Parkinson's Disease (PD), Multiple System Atrophy (MSA), Progressive Supranuclear Palsy (PSP) and Dementia with Diffuse Lewy body (DLB). Despite having diverse symptoms, they are usually progressive, with increasing in severity over time [28]. Movement disorders drastically impact a persons ability to be self sufficient in day to day life, although they are not usually life threatening.

## 1.1 Parkinson's Disease - Importance, Causes and Symptoms

PD had affected about 6.2 million people globally in 2015 [136]. Since then, the number is estimated to have risen to around 10 million [128], making it the second most widely occurring neuro-degenerative movement disorder, only behind AD [4]. PD is present in 22 out of 100,000 person-years when considering all age groups [53], [138]. Person years take into account both the number of people in an experiment as well as the time each person spends being a part of that experiment. For instance, a study with 1000 participants that ran for 1 year would generate 1000 person years worth of data [23].

PD is usually much more prevalent in aging people, being incredibly rare in subjects younger than the age of 50 [103], with an estimated 96% of all cases being found in patients older than 50 and biological males being almost 1.5 times more likely to suffer from it compared to those who are female [1]. PD is found in 529 out of 100,000 person-years for the elderly population [53], [138]. Above 1% of the worlds population aged above 60 and around 4% at the age of 80 are affected by it [4].

With the advancement in modern medical sciences, the life expectancy of people has increased significantly. Paul et al. [123] estimated in 2013 that the number of people older than 60 or older will be higher than the number of children younger than 5 years by 2018. In 2019, the World Bank stuff estimates the population with ages above 65 to be more than 9% of the world's population, the highest ever, based on age/sex distributions of United Nations Population Division World Population Prospects [97]. With this vast and ever increasing number, there arises a need for development of health care system targeting the aging population susceptible to PD. However, it was observed that in subjects without PD older than 80, there was no major decline in midbrain Catecholaminergenic neurons, which indicates that PD is not simply a byproduct of the natural aging process [59]. The incidence of PD was actually found to be in decline after the age of 70 to 75 years [57], [132], [133], and PD is not one of the major causes of mortality in people who are older than 85

years [3]. So it can be concluded that, while PD is sporadic in the younger people and mostly affects the elderly, age itself is not one of the reasons for it.

The main cause of PD is thought to be the loss of nerve cells or neurons in the Substantia Nigra (SN) region [32], [76], [103], [138]. SN is located in the Basal Ganglia (BG) of the human midbrain, playing a significant part in movement and coordination. Neurons in this part of the brain are responsible for producing an catecholaminergenic organic chemical known as dopamine, which acts as a neurotransmitter in the brain by helping the neurons communicate, facilitating the coordination of body movements. If the amount of Dopamine produced in the brain is insufficient, communication between neurons for coordinating body movement is hampered; leading to PD.

PD has been clinically defined and studied for decades, but the exact mechanisms leading to it are still unclear [95]. PD is characterized by a number of neurological and motor symptoms. Since the loss of Dopamine causes PD, the motor symptoms are more prominent and manifest earlier including autonomic dysfunctions, resting tremors, rigidity and stiffness of trunk and limbs, Bradykinesia, Dyskinesia, irregular stride length and gait speed, akinesia, Freezing of Gait (FOG), falls and postural disorders [67]. The motor symptoms for PD usually incapacitate a subject, creating difficulties in sitting and standing up. The patient also suffers from losing the normal pendulum motion of the arms and displaying very small steps [32], [138]. The motor symptoms are followed by subsequent non motor and neurological problems like speech impediments, olfactory dysfunctions, sleep, cognitive and mood disorders, fatigue [131] with dementia being common in the advanced stages [1].

## 1.2 Diagnosis of Parkinson's and Challenges

Although neuroimaging and genetics have developed rapidly in recent years, PD is primarily diagnosed pathologically [12]. It is difficult to diagnose without the manifestation of motor symptoms, which are often unlikely to appear before 50% to 70% of the total neurons in the brain have been damaged [17], making it extremely difficult to administer any kind of preventive measures.

Additionally, the symptoms for PD in the early stages resemble other medical conditions like Parkinsonian Variant of Multiple System Atrophy (MSA-P), PSP, ET; often causing misdiagnosis [108]. Pagal et al. [91] discovered that PD had a misdiagnosis rate of 24% depending on the person performing diagnosis strictly following clinical guidelines. Primary care doctors had a successful diagnosis rate of 53% and movement disorder experts had a rate of 75%. There are no established laboratory test or blood analysis for diagnosing PD, it is done by interviews and observations with the patients, which should be conducted by a neurologist well versed in movement disorders to avoid misdiagnosis. This process is heavily reliant on expertise and is subjective. There is also possibility of human error and bias affecting the diagnosis. The limited number of specialists also have to allocate a lot of their time in assessing large quantities of data, which is time and resource consuming.

Although a guaranteed cure for PD has not been found yet, early detection might play a crucial role in slowing or stopping the progression of the disease. Some new forms of treatment like Exenatide [9] show promising results in case of early detection. So developing methods capable of identifying PD in its early stages remains a priority.

## 1.3 Computer Aided Diagnosis of Parkinson's Disease

Analyzing the structural changes in the brain using Medical Imaging techniques have been proven to be helpful for detecting neuro degenerative diseases with cognitive impairments recently [100], [109]. Significant research has been conducted on diagnosing PD with the help of Computer Aided Diagnosis (CAD) techniques. It is easier to capture and detect motor-based symptoms compared to non-motor symptoms with computer detection techniques. Over the years, different types of technologies have been used depending on the nature of the symptoms. Voice tremors corresponding to Dyskinesia is captured with the help of microphones [65]. Tremor in the limbs, also corresponding to Dyskinesia is monitored with motion sensors placed in the arms and wrists [96].

FOG corresponding to Akinesia is monitored with the help of motion sensors as well as image and video of the patients [11]. In particular, Magnetic Resonance Imaging (MRI) provide better performance in brain structure analysis because of having high contrast and resolution within soft tissue.

The PD detection algorithms are usually complicated due to their nature of identifying peculiar symptoms, being heavily reliant on hand crafted features, which are time consuming and extremely sensitive to outliers in the data. The features are needed to identify each special symptom. The complex nature of these algorithms is a hindrance in the development of real time PD detection systems [91]. The delays caused by these issues might result in increase in the progression of the disease or worsen the symptoms. Therefore, systems capable of detection of PD and its symptoms with high accuracy without complex handcrafted feature engineering are a necessity.

## 1.4 Freezing of Gait (FOG) - Major Symptom of Parkinson's

Studies show that around 60.5% of all PD patients experience a minimum of one fall and 39% of all patients experience recurrent falls, which might lead to fracture as well [6]. Falls and fractures significantly damage the quality of a persons life, might lead to disabilities and has a 10.6% chance of being fatal [53]. Falls can be one of the usual aftermath of FOG, which is one of the most common symptoms of PD, with around 50% of all PD patients being affected [30], [87]. A typical FOG causes the patient suddenly experiencing a sudden inability to move, which often occurs while initiating gait, making turns while walking, when experiencing stress, approaching narrow spaces or performing multiple tasks in parallel [86], [122]. The patients report a feeling of their feet being glued to the ground during these events [84], [118]. FOG episodes are transient and they usually last for a few seconds, but can last for upto 1-2 minutes in some cases [26], [84]. Based on the signals received from sensors worn around the ankles, it was found that while normal walking steps occur at a frequency of 0.5 Hz to 3 Hz, FOG exhibits a frequency of 6 Hz to

8 Hz [5], [56], [79].

Typically, FOG is very difficult to estimate and predict, but it can cause falls and pose health risk for the affected elderly [55], [62], [77], [86]. An accurate prediction and detection of FOG can reduce accidents and thus improve the quality of life of the patients and their loved ones. However, FOG is one of the least responsive to medical treatments among PD motor symptoms [26]. External cues leading to auditory or visual stimulus show promising results as treament for FOG. One such treatment is to use Rhythmic Auditory Stimulation (RAS) [38], which produces a rhythmic ticking sound to help the patient resume normal gait when a FOG is detected. Cueing on demand is more efficient than continuous cueing in reducing FOG duration. Continuous cueing would be disruptive in the patients day to day life. There is also the possibility that it would lose effectiveness because the patient will be de-sensitized to the cues if it is continuous. So the goal is to automatically detect FOG to trigger the cueing only when it is needed.

Current methods for detecting FOG mostly consist of movement tests in controlled lab setting, self-monitoring and assessment by patients, manual video analysis by professionals and detailed Freezing of Gait-Questionnaires (FOG-Q) used to assess the frequency and severity of FOG episodes and symptoms [29], [31], [82], [106]. Although somewhat accurate, these methods suffer from shortcomings because of their clinical setup not reflecting real-world scenarios. FOG events usually tend to occur at home or while the patients are performing Activities of Daily Living (ADL) [83], [118], [131], which are different from the clinical test setup. Furthermore, FOG-Q is usually dependent on the opinion of the patient, which might be subjective and biased. In addition, since PD patients are prone to experiencing dementia and memory loss, their self-assessments are unreliable [106]. Assessments are also time consuming and not suitable for continuous monitoring the the patients health.

With the advancement in technology, using wearable sensors to monitor movements, body temperatures, heart rates and other physical parameters has become increasingly commonplace [117]. These sensors are lightweight, comfortable and usually do not hamper a person's daily activities while mon-

itoring ADL. The data recorded from wearable sensors for activity detection has brought promising performance in various applications, especially when combining with modern Machine Learning (ML) and Deep Learning (DL) based techniques [39], [61], [85], [88], [137]. There are wearable sensors that use auditory stimulation to treat FOG, which help shorten the duration of the events [11]. But these sensors cannot effectively help stop FOG episodes because of the latency of detection, which can still be hundreds of milliseconds in the best case scenarios [52].

There have been many applications using wearable technologies along with ML and DL based techniques to monitor motor functions of PD patients, aiming to achieve more effective treatment and reduce healthcare expenses [60], [71], [114]. These approaches can provide an unobtrusive and comfortable experience to the patient, while collecting personalized long term relevant medical history and improving the quality of treatment. Maetzler et al. [69] state that an automatic FOG analysis and detection system could play a vital role in monitoring the occurrence and evolution of FOG events over time. Although a permanent and guaranteed cure for PD or FOG itself not been available at this time, a sufficiently accurate automatic monitoring system might prove to be helpful in minimizing the frequency and duration of FOG events. Rhythmic auditory cues RAS have shown to improve walking by maintaining the speed and amplitude of movements [7], [64], [105].

## 1.5 Application of Deep Learning for Parkinson's Analysis and Challenges

With the popularity of these sensors, the amount of available data is increasing at a rapid pace, which facilitates the use of DL based techniques. DL is under the scope of Artificial Intelligence (AI) that has the capabilities to automatically extract features from data without manual feature engineering. DL based end-to-end classifiers have shown promising performance, outperforming ML based classifiers in general, if sufficient amount of training data is available [47]. Recently DL based approaches have been adopted to perform

tasks related to Human Activity Recognition (HAR) using data from various sensors. [39], [85], [88], [137].

Deep Convolutional Neural Network (CNN) are a type of common DL architectures. Lecun et al. [63] mention that CNN can be applied to temporary signals and images to automatically extract abstract distinct features by combining several convolutional operators. Although CNN are proficient in extracting invariant local features from data, this architecture often falls short when the data has global time dependency, which is often the case with data obtained from wearable sensors. Recurrent Neural Network (RNN) are able to solve this issue because the connections between the nodes of this architecture exhibit a discrete-time dynamical system [54], [92]. Long Short Term Memory (LSTM) is one of the most widely used RNNs, able to model time dependency in sequential time series data using various logic gates to control a memory space [41].

Neural Network Ensembling is the learning paradigm of training a collection of neural networks to perform the same task [125]. The idea of ensembling was introduced by Hansen et al. [37], who proposed that the generalization ability of a Neural Network based system can be significantly improved by training a number of neural networks and by combining their solutions to solve the same problem. A typical ensemble architecture consists of two steps, i.e., training multiple components or constituent neural networks, and then creating an architecture that combines their outputs. In recent years, ensemble learning techniques have been applied to PD detection tasks and they have achieved significant success [8], [81].

One of the major challenges to applying DL based techniques is data scarcity. A large number of properly labeled training data is necessary in order to train a DL based model. The small size of the training dataset is responsible for the poor performance of many ML and DL based models. With medical issues such as detecting PD and monitoring its symptoms, such data are hard to obtain. There are issues related to privacy of the patients, lack of domain experts to properly label the dataset, discrepency among data collected from different patients or different sensors, lack of time and resources

for data collection etc. There are multiple ways to solve such issues such as data augmentation, generating synthetic data or transfer learning.

## 1.6 Contributions

The total contribution of this thesis can be organized in two main categories.

- PD Detection from Neuroimaging Data.
  - Using transfer learning based approach with ImageNet Large Scale Visual Recognition Challenge (ILSVRC) models considering 2 situations, trained on ImageNet [20] and without any prior training to detect PD from weighted T1 MRI brain scans.
  - Propose multiple ensemble architectures for PD detection combining the ILSVRC models.
  - Compare the performance of using Whole Brain Scans, extracted GM and WM with and without smoothing applied using multiple evaluation metrics.
  - Perform Occlusion analysis to identify the regions of the brain that have a large significance in the decision making process of our models and present relevance per brain area graphs.
- FOG detection and prediction for PD patients
  - Propose 2 Neural Network (NN) architectures with various time frequency representation techniques from publicly available accelerometer data including non overlapping windows, handcrafted features, RP, STFT, DWT and PWVD and thoroughly evaluate the performance of proposed models with their respective data modalities.
  - Develop three ensemble architectures combining the proposed models and various data modalities and evaluate their performance.
  - Select appropriate sensors for data collection from an experiment conducted by A. T. Still University and apply our model trained on public dataset to the new data.

## 1.7 Organization of the Thesis

The rest of the thesis is organized as follows:

- Chapter 2 : Literature review for both PD detection from MRI images and FOG monitoring are presented in this chapter. The definition and formulas for the multiple evaluation criteria used in this thesis are also provided here.

- Chapter 3 : In this chapter, the proposed methods for PD detection from MRI are detailed, including dataset details, model architectures, performance evaluation etc. This chapter also contains results for Occlusion analysis on some of our models.

- Chapter 4 : This chapter contains work on FOG detection and prediction using various modalities of data generated from accelerator sensor signals. The chapter contains details of the dataset, preprocessing and feature extraction steps, model architectures, results etc. This chapter also presents the application of the proposed models on data collected from patients in a study performed by A. T. Still University.

- Chapter 5: We list our conclusions and indicate future direction for our work in this chapter.

# Chapter 2

# Background Material

Significant research has been conducted in PD diagnosis and progress analysis using various data modalities including human voice, motion and neuroimaging data.

## 2.1 Background and Literature Review for Parkinson's Disease detection

Over the years, a multitude of ML [27], [94], [95], [110] and DL [18], [24] based approaches have been introduced for the detection of PD. Focke et al. [27] extracted GM and WM from Magnetic Resonance (MR) images and fed them to a Support Vector Machine (SVM) Classifier for PD detection, achieving 39.53% and 41.86% classification accuracy using GM and WM respectively. Radial Basis Function Neural Network (RBFNN) was used by Pazhanirajan et al. [94] for PD classification. Babu et al. [116] achieved a 87.21% accuracy in classifying PD using GM with a CAD system. They identified Superior Temporal Gyrus (STG) as a potential biomarker that plays a vital role for PD.

Choi et al. [18] achieved an accuracy of 96% for PD detection using Single Photon Emission Computed Tomography (SPECT) imaging with CNN. Although their accuracy was very high, SPECT Imaging is invasive and not very popular as it requires injecting a radioactive tracer into the patient. Around 100 times more MR scans were performed compared to SPECT over one year period in the National Health Service (NHS) operation in England. Thus the

SPECT approach is less practical for normal medical use due to limited sample size, despite its reported high accuracy. Also, their dataset is class imbalanced since about 69% of the data is from PD patients. Class imbalance causes the models to over classify the majority class [13].

Detecting PD from resting-state functional MRI (rsf-MRI) aims to discover subtle changes in blood oxygenation level. For detecting PD, researchers focus on using structural MRI (sMRI) in order to capture the anatomical details. Long et al. [66] used a ML based approach and they achieved 87% classification accuracy, but the dataset used by them was very small. Rana et al. [99] used a SVM for classification with t-test feature selection on WM, GM and Cerebrospinal Fluid (CSF) achieving 86.67% accuracy for GM and WM, and 83.33% accuracy for CSF. In another work [101], the authors used the relation between tissues instead of considering the tissues separately and achieved an accuracy of 89.67%.

Among the various regions in the brain, the SN region has significant correlation with PD according to Braak's neuroanatomical model of Parkinson's Disease [12] and it is often used as a Region Of Interest (ROI) in PD identification. However, the challenge is the lack of brain MR images for GM and WM training.

To address this issue, we explore the feasibility of pre-training a model with non PD related images. ImageNet [20] is one of the well known image datasets for Computer Vision (CV). It is unrelated to PD. ImageNet is organized according to the WordNet [21], [139] hierarchy. WordNet is one of the largest lexical database of English words with nouns, verb, adjectives etc. organized into "Synonym Sets" or "synsets", which are sets of cognitive synonyms. A "synset" describes a meaningful concept with multiple words or word phrases. WordNet contains more than 100,000 synsets, with more than 80,000 being nouns. Currently ImageNet labels images using only the nouns from WordNet. Each node of WordNet hierarchy is represented by a thousand image samples in ImageNet on average. The ILSVRC [107] evaluates the performance of various algorithms for object detection and image classification on the ImageNet dataset. The challenge has 1000 object categories, with the

categories containing both internal and leaf nodes of ImageNet, but they do not overlap with each other. It is to be noted that the WordNet hierarchy contains more categories, but these 1000 non overlapping classes were chosen. Figure 2.1 shows two sample images from the ImageNet dataset and their positions in the WordNet hierarchy. Kornblith et al. [58] proposed that models performing well on the ILSVRC also perform better when they are applied on other datasets.

(a) Animal-Beast-Chordate-Vertebrate-Mammal-Placental-Carnivore-Feline-Big Cat-Lion

(b) Artifact-Instrumentation-Container-Wheeled Vehicle-Self Propelled Vehicle-Motor Vehicle-Car/Automobile-Race Car

Figure 2.1: Sample Images from ImageNet [20] dataset and their position in the WordNet [21], [139] Hierarchy

## 2.2 Background and Literature Review for Freezing of Gait Detection and Prediction

Smart sensors have been commonly used as a tool for assessing motor symptoms such as FOG in PD and other movement disorders. This is possible because of the improvements in computational power of small devices [14]. Existing FOG assessment methods using these sensors can be categorized into different groups depending on the sensor types, sensor locations, extracted features, and the analytics methods. FOG detection can be conducted real-time [126]. However, FOG detection and prediction are challenging tasks because of the variability of event duration and frequency. We observe that previous studies mainly captured FOG episodes that are not consistent with the patients' normal daily activities because their data were simulated in laboratory settings. In this section, we review related work on FOG detection.

An early FOG detection method was proposed by Han et al.[36] using Activity Monitoring System (U-AMS) based on wavelet power features for discrimination of abnormal movements in PD patients which showed a promising avenue for research. Moore et al. [80] then proposed a threshold based method for FOG detection by defining the Freeze Index (FI), which is the ratio between the power of the signal in "freeze" band (3-8 Hz) divided by the power of the signal in the "locomotion" band (0.5-3 Hz). The proposed method marks FOG episodes when FI exceeds a certain threshold. The subject dependent experimental results, which means training and testing separate instances of the model on data from each patient, show 78% correct detection of FOG (true positive rate) and 20% false positive rate. Bachlin et al. [10] presented a real time FOG detection method by introducing a new term to Moore et al. [80] method, called Power Index (PI), which is the addition of Walking Band (WB) and Freezing Band (FB) that indicates the movement amount. In [11], FOG episodes are determined using two thresholds (Freezing Threshold (FTH) and Power Threshold (PTH)) given $FI > FTH$ and $PI > PTH$ respectively. In this method, once the FOG episodes are detected, the patient will get the auditory signals until his normal walking ability is resumed. They reported

73.1% and 81.6% for sensitivity and specificity respectively. The author also created the Daphnet data set [11] for FOG assessment methods evaluation.

The first proposed FOG detection method based on ML was by Mazilu et al. [74]. The features for this classification were from the work of Bachlin et al. [11] with some additional features including mean, standard deviation, entropy, energy, FI and power of the acceleration signals. Random Forest (RF), Naive Bayes and K-Nearest Neighbour (KNN) were the ML algorithms for doing classification. Motion data capture was done by a smartphone and a wrist acceleration sensor. The best obtained results were 66.25 and 95.83 for sensitivity and specificity respectively with RF using 10-fold cross-validation. In the following year, they presented another automatic FOG detection system using wearable sensor. In this work, they did multi-class analysis as the "**PreFOG**" motion was considered a new class (FOG vs. PreFOG vs. normal locomotion). Learning was conducted by studying the time domain and statistical features from the motion data. In this new work, they could improve F1 score by 8.1%. The new automatic FOG detection method introduced auditory cueing to warn the patient about FOG episodes. In the same year (2013), a system for automatic FOG detection was proposed by Tripoliti et al. [135]. The system was based on four steps: data imputation (interpolation), band-pass filtering, entropy calculation, and automatic classification (Naive Bayes, RF, Decision Trees and Random Tree). Data was obtained from 5 healthy subjects, 5 PD patients with FOG symptoms, and 5 PD patients without FOG symptoms. The results show 81.94% sensitivity, 98.74% specificity, 96.11% accuracy and 98.6% Area Under Curve (AUC) using RF. Another proposed FOG detection work in 2013 was by Moore et al[78], which assesses seven sensors placed in different locations for gait analysis. Their analysis found that the shank and back were the most convenient places for the sensors producing best results. However, they found that using all the seven sensors could get higher and more robust performance with sensitivity 84.3% and specificity 78.4%.

In 2015, Zack et al.[141], presented a threshold based FOG detection following the approach of Moore [78] using a single triaxial accelerometer placed at the waist. Receiver operating characteristic (ROC) curves were drawn to

determine a global FI threshold to distinguish between FOG and non-FOG episodes for different tasks. In addition to the global FI threshold, they calculated the sensitivity and specificity of the FI threshold for each subject. Combining all task results, a sensitivity of 75% and specificity of 76% were achieved [126].

Rodríguez et al [106] presented a novel approach for FOG detection using machine learning techniques and daily activities of the PD patients in real environments. They extracted 55 FOG related features from 21 PD patients using just a single waist-worn triaxial accelerometer. SVM with leave-one-out cross-validation was used for classification in two scenarios: user independent and user dependent. Experimental results show a sensitivity of 88.09% and specificity 80.09% with R-10-fold cross-validation, and a sensitivity of 79.03% and specificity of 74.67% for Leave-One-Subject-Out (LOSO) evaluation. After that, Sama et al [111] decreased the number of features from 55 to 28 28 for the same data set. The extracted features were sent to 8 different classifiers with greedy subset selection process, 10-fold cross-validation and different window sizes. The results of FOG detection at patients' homes were 91.7% and 87.4% for sensitivity and specificity respectively, which are better than the results of Rodrigues's method.

Orphanidou et al [89], evaluated ML algorithms to identify the FOG prior to its onset. An accelerometer time series dataset containing 237 individual FOG events from 8 patients identified by experts was considered, from which features were extracted and presented to 7 machine learning classifiers. SVM achieved the highest performance in comparison with the benchmark techniques. The classification algorithm was applied to 5 second windows using 18 features, obtaining balanced accuracies (the mean value of sensitivity and specificity) of 91%, 90%, and 82% over the Walk, FOG and Transition classes, respectively. However, the need for systematic analysis of the problem was identified. Therefore, in their next study [90], they specifically focused on the early detection of a FOG event, through classification of the transition class using varying size time windows and time/frequency contrary to the majority of previous studies that recognized FOG only when it had occurred. In their

paper, the Daphnet dataset [11] was used with accelerometer signals obtained from sensors mounted on the ankle, thigh and trunk of the PD patients.

Data augmentation was performed on the dataset to include another class label called 'transition' that showed the episodes before FOG occurrence. Daphnet features were sent out to a group of 5 classifiers, including Gradient Boosting (GB), Extreme Gradient Boosting, SVM, RF, and NN. Experimental results show that SVM with Radial Basis Function (RBF) kernels has the best performance with sensitivity of 72.34%, 91.49%, 75.00%, and specificity values of 87.36%, 88.51% and 93.62%, for FOG, transition and normal activity classes, respectively.

DL techniques have also been used for automatic FOG determination. DL can handle multi-modal data, missing information and high dimensional feature spaces. The first proposed FOG detection method using DL was by Camps et al. [15]. The proposed 1D CNN has 8 layers, which is trained using a novel spectral data representation strategy that considers information from both the previous and current signal windows. The data was collected from 21 subjects, consisting 9-channel signals recorded from a waist-worn Inertial Measurement Unit (IMU) with three tri-axial sensors: accelerometer, gyroscope, and magnetometer. The experimental results show a performance of 90.6% for the Geometric Mean (GM), an AUC of 0.88, a sensitivity of 91.9%, and a sensibility of 89.5%.

In 2019, San-Segundo et al. [113] presented a study to evaluate the robustness of different feature sets and ML algorithms for FOG detection using body-worn accelerometers. They used four feature sets: (Mazilu et al. [74] features, HAR features, Mel Frequency Cepstral Coefficients (MFCCs) features, and Speech Quality Assessment (SQA) features). They also used four classification (RF, Multi-Layer Perceptron, Hidden Markov Model (HMM), and Deep Neural Network (DNN)). Evaluation was performed using a LOSO cross-validation. The best results were obtained when using the current window and three previous windows, with the feature set composed of Mazilu features [74] and MFCCs [112]. They found that the best classifier was a deep CNN achieving an AUC of 0.93 and an Equal Error Rate (EER) of 12.5%.

In 2020, Sigcha et al. [121] evaluated some ML and DL classification and detection techniques with accelerometer signals acquired from a body worn IMU to enhance the FOG detection performance in real world home environments. Three data representations proposed in the literature were reproduced (including Mazilu features [74], MFCCs [112], and Fast Fourier Transform (FFT)) to establish a baseline using RF classifier with 10-fold cross-validation (R10fold) and LOSO. This analysis was also conducted to find the best data representation to test DL approaches including: a Denoiser Autoencoder, a DNN with CNN, and a combination of CNN and LSTM layers. For comparison proposes, shallow algorithms such as One-Class Support Vector Machine (OC-SVM), SVM, AdaBoost, and RF were tested. OC-SVM was set up to only identify the important class, FOG in this case. This study was evaluated on the data collected by Rodríguez-Martín et al. [106], which includes recordings from 21 PD patients, who manifested FOG episodes when performing ADL at their homes. The best performance for AUC was 0.93. Their results illustrate that modelling spectral information of adjacent windows through an LSTM model can improve the performance of FOG detection without increasing the length of the analysis window.

## 2.3 Performance Evaluation Criteria

Multiple evaluation metrics were used to properly quantify the performance of our proposed methods. This was necessary because detection accuracy by itself is not always a reliable evaluation metric. For example, in cases where a majority class dominates the dataset, it might be possible that the detection accuracy is very high despite the model failing to identify the minority classes. To ensure that the performance of our models were properly evaluated and it is not over classifying the majority class, a number of evaluation metrics were utilized in our study.

### 2.3.1 Detection Accuracy

Detection accuracy is the most widely used evaluation metric. It is defined as the fraction of predictions by a model that are accurate. Detection accuracy can be computed as Eq. (2.1). The output of this metric ranges between $(0, 1)$, with 0 being completely inaccurate and 1 representing perfect prediction.

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ records} \tag{2.1}$$

#### 2.3.1.1 Precision, Recall/Sensitivity, Specificity, $F_\beta$ Score

Precision, Recall/Sensitivity, Specificity and $F_\beta$ Score are very important in understanding the performance of a model. For multi-class classification, each of these metrics computes an individual class and then their weighted average is calculated.

Precision for a class is the measure of the classifier's ability to not classify a negative sample as positive, as defined in Eq. 2.2.

$$Precision(A_k, B_k) = \frac{|A_k \cap B_k|}{|A_k|} \tag{2.2}$$

Recall/Sensitivity of a class measures how well the classifier can identify positive samples of a class, as defined in Eq. 2.3.

$$Recall(A_k, B_k) = \frac{|A_k \cap B_k|}{|B_k|} \tag{2.3}$$

Specificity for a class is defined as the ability of a classifier to reject samples that are not a member of that class.

$$Specificity(C_k, D_k) = \frac{|C_k \cap D_k|}{|D_k|} \tag{2.4}$$

The $F_\beta$ is calculated as the weighted harmonic mean of Precision and Recall, ranging between [0,1]s, with 1 being the best possible value, as presented in Eq. 2.5.

$$F_\beta(A_k, B_k) = (1 + \beta^2)\frac{Precision(A_k, B_k) \times Recall(A_k, B_k)}{\beta^2 Precision(A_k, B_k) + Recall(A_k, B_k)} \tag{2.5}$$

where,

- $A_k$ is the predictions for class k
- $B_k$ is the occurrences for class k
- $C_k$ is the predictions for samples not in class k
- $D_k$ is the occurrences for samples not in class k
- $k$ represents a class in range $1 : K$, K being the number of classes

### 2.3.2 Matthews Correlation Coefficient

Matthews Correlation Coefficient (MCC), also known as Phi Coefficient, was proposed by Matthews et al. [72] in 1975. MCC offers a balanced measure of quality for both binary and multi-class classifications, which can be used even if the classes are imbalanced. The value of this metric ranges from $[-1, +1]$. A MCC value of $+1$ indicates perfect prediction, 0 indicates random prediction and $-1$ indicates inverse predictions. Gorodkin et al. [34] generalized MCC for multiple classes as the $R_K$ statistic, defined with respect to confusion matrix C for K classes following Eq. (2.6).

$$MCC = \frac{c \times s - \sum_k^K p_k \times t_k}{\sqrt{(s^2 - \sum_k^K p_k^2) \times (s^2 - \sum_k^K t_k^2)}} \tag{2.6}$$

[75] where,

- $t_k = \sum_i^K C_{ik}$, the number of occurrences of class $k$
- $p_k = \sum_i^K C_{Ki}$, the number of predictions for class $k$
- $c = \sum_k^K c_{kk}$, total correct predictions
- $s = \sum_i^K \sum_j^K Cij$, total number of samples

# Chapter 3

# Parkinson's Disease Detection from Magnetic Resonance Imaging

## 3.1 Introduction

As mentioned in Section 1.2, the existing methods for PD detection comes with a number of limitations including being influenced by human errors and biases, being heavily reliant on individual expertise as well as being time and resource consuming for the patients [101]. Advanced neuroimaging techniques have resulted in significant improvements in neurodegenerative disease diagnosis [1.3]. One of the most widely used neuroimaging techniques is MRI, which is comparatively inexpensive, non-invasive and capable of generating images of the soft tissue with high contrast and resolution [1.3]. MRI additionally has the ability to identify sub-cortical volume and shape variation in the brain [140]. In this chapter, we analyze whether transfer learning based approach is suitable for identifying PD from MRI scans. Models designed for ILSVRC [107] were used both with previous training weights from Imagenet [20] dataset and without any training. Our approach was different from ordinary transfer learning based approaches in that we did not use a related dataset for our training. Multiple ensemble architectures were also proposed combining the models and their performance was evaluated. Finally occlusion analysis was performed to quantify the importance of the regions of the brain in the decision making process of the model.

## 3.2 Data

For this experiment, we used Parkinson Progression Markers Initiative (PPMI) dataset [134], a comprehensive set of clinical, imaging and bio-sample data defining PD progression and diagnosing bio-markers. The dataset consists of T1-weighted sMRI scans for 568 PD and Healthy Control (HC) subjects from which 445 subjects were chosen and the rest were discarded due to some structural anomalies during the preprocessing steps. The resulting data had a class imbalance with 299 PD and 146 HC subjects. The HC subjects were people over 30 years old without PD who signed up for the study, and were without a first degree blood relative with PD. To address this issue, 153 HC T1-weighted sMRI scans from the publicly available IXI dataset [46] were collected. Our final dataset was class balanced with 598 subjects. The demographic for the dataset is presented in Table 3.1.

Table 3.1: Demographic Data

| | PD | HC | Average |
|---|---|---|---|
| Age (Years) | $62.0 \pm 9.54$ | $49.2 \pm 16.9$ | $55.6 \pm 15.1$ |
| Sex (Male / Female /Total) | 189 / 110 / 299 | 172 / 127 /299 | 361 / 237 / 598 |

From Table 3.1, it can be observed that there was a difference between the mean age of our PD and HC subjects. The mean age for PD patients is 62 years whereas the mean age for HC subjects is around 49.2 years. But we believe that this difference does not influence the analysis, based on existing research that the human brains are more or less structurally developed by the age of 30 [51]. After that the brain ages, but there is no structural development. According to Cole et al [19], as our brains age, we tend to experience cognitive decline and are at greater risk of neurodegenerative disease and dementia. Figure 3.1 presents an illustration of the concept of brain aging trajectories. With increase in age, even healthy people face a higher risk of cognitive impairment and brain diseases, eventually crossing a threshold for appearance of symptoms. The trajectories differ from person to person. The

blue line shows the trajectory for a person with genetic or developmental environmental factors that confer a higher rate of aging throughout life. A person may experience a traumatic injury or infection as an adult shown by the black arrow, which would result in them following the accelerated purple trajectory or accentuated, but stable yellow trajectory of brain aging. But Cole et al.[19] show that even a person with generic or developmental environmental factor that contributes to rapid aging (blue line) or someone with an injury or infection causing accelerated aging (purple line), will reach the symptomatic threshold for cognitive impairment or brain diseases after 60 years. A normal person with healthy aging will reach this threshold much later in life. This shows that using our PD and HC patient data with an average age of 55.6 years is suitable for comparison and analysis purposes.

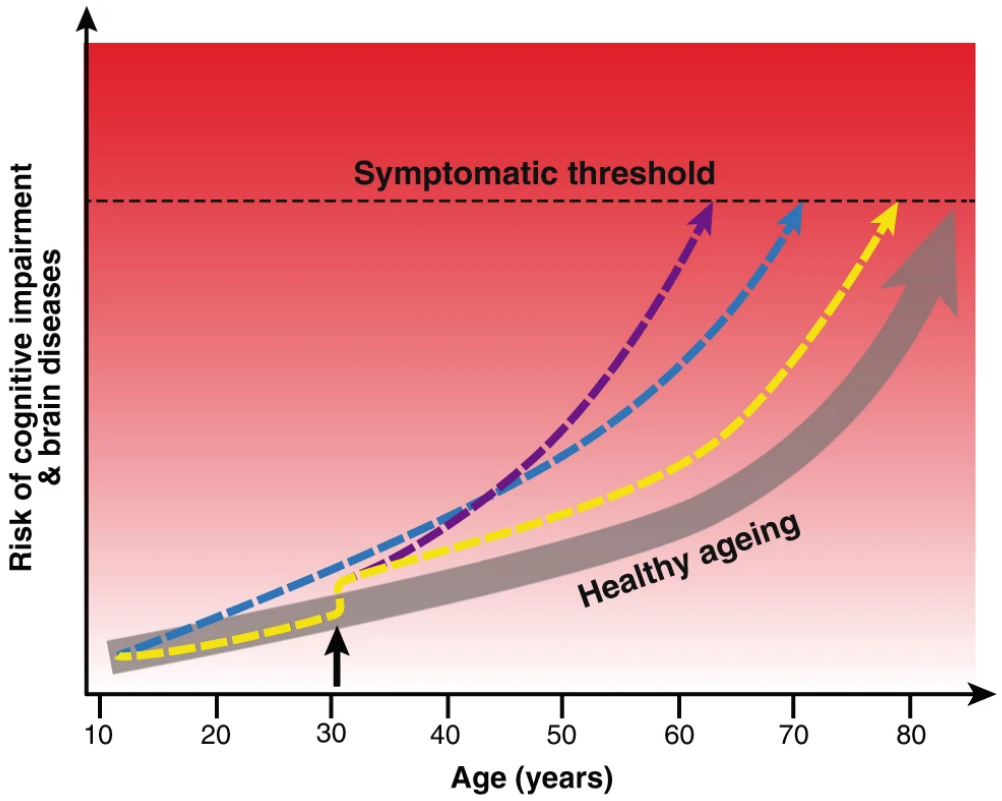



Figure 3.1: Brain aging with age represented as trajectories. The thick brown line represents healthy aging. The purple line represents rapid brain aging due to injuries and blue line represents brain aging due to generic or environmental issues. The yellow line represents more prominent but stable brain aging due to injuries. X axis represents age and Y axis represents probability. Figure taken from [19]

.

## 3.3 Preprocessing

Preprocessing is one of the essential steps in any CAD system, specially in case of neuroimaging data analysis. There are morphological and dimensional differences between our data since the scans come from different machines. To make the data comparable we had to standardize it to a common format. All scans were resized to the same dimensions. For preprocessing, Statistical Parameter Mapping (SPM12) [2], [127] and Computational AnatomTtoolbox (CAT12) [130] were used.

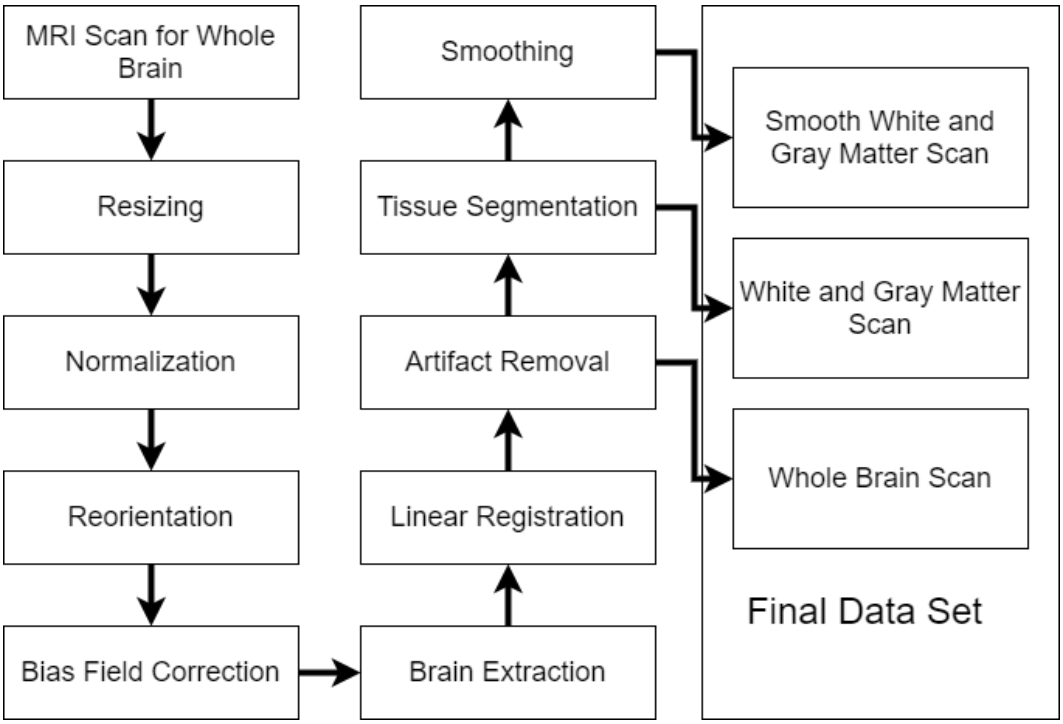
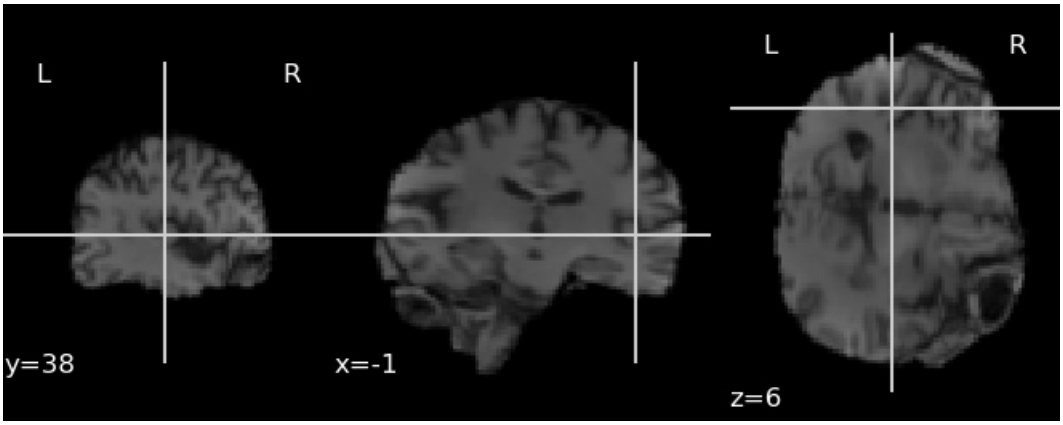


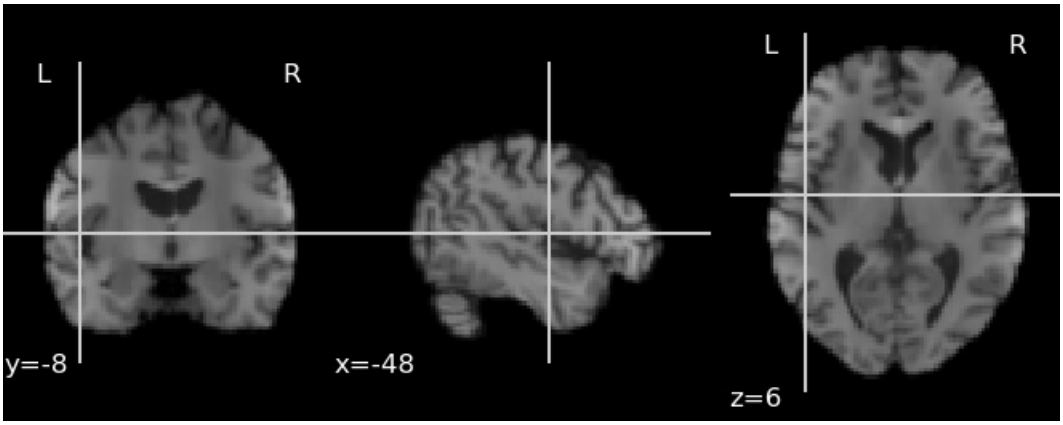

Figure 3.2: Preprocessing Pipeline

Figure 3.2 shows the structure of our preprocessing pipeline. MRI intensity varies from subject to subject. To minimize discrepancies we normalized the values to [0,1] range. Then all images were aligned to a standard space named Montreal Neurological Institute (MNI). Then a bias field correction (FAST) [142] is performed to remove general intensity non-uniformities. FNIRT / BET [48] was used to extract brain from the scans removing the skull, fat and background regions which do not contain useful information. The data was registered to MNI152 format (FLIRT) [49], [50]. After that artifact removal was performed, i.e. any voxel intensity values higher than 1 is corrected to be in the range [0,1]. Then a deformation method was applied to extract GM and WM from the scan and a 8mm Isotropic Gaussian Kernel was used to smooth and increase the signal-to-noise ratio and remove unnecessary portions of the scan. GM and WM are significant in brain structure analysis and can help in PD identification. Finally we have three separate datasets: whole brain scans, GM and WM extracted from the brain and Smoothed GM and WM. An example of the extracted brain is shown in Figure 3.3 and the

resultant WM and GM extracted from the brain is given in Figure 3.4 and Figure 3.5.

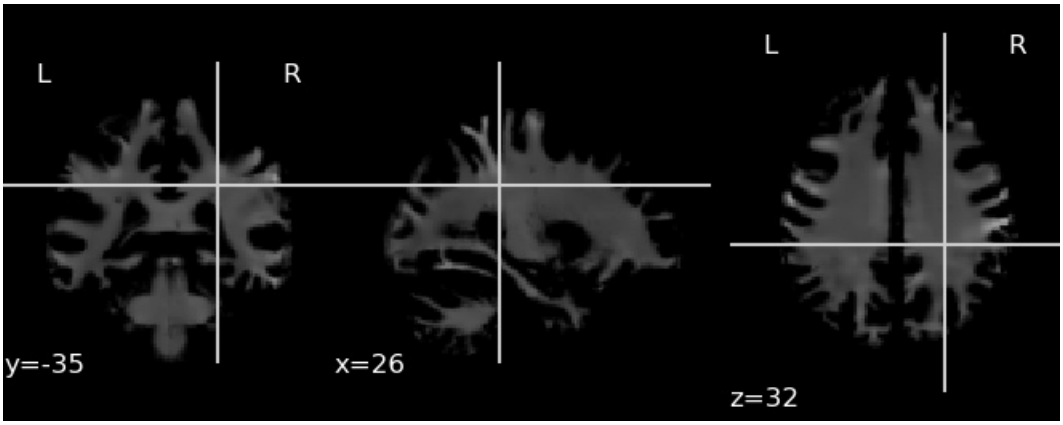



(a) Whole Brain Scan for Healthy Control subject

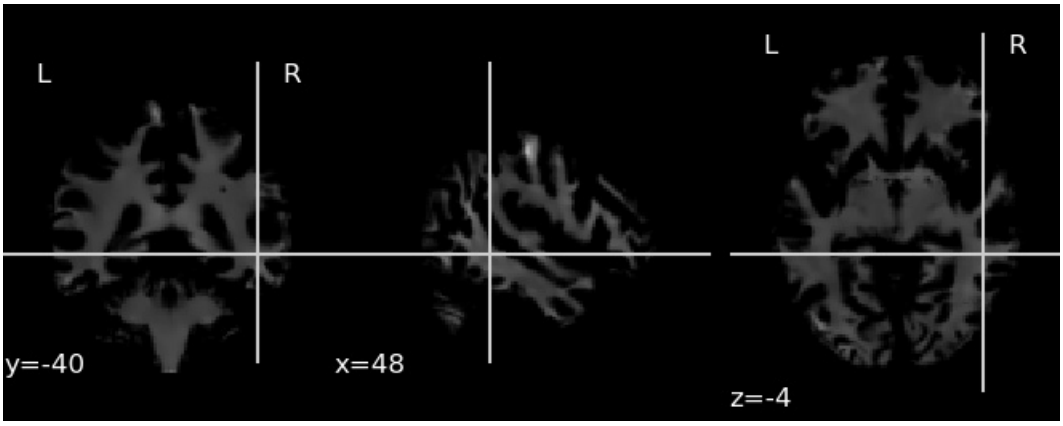



(b) Whole Brain for Parkinson's subject

Figure 3.3: Sample MRI scan comparison for a Healthy Control subject and Parkinson's Patient

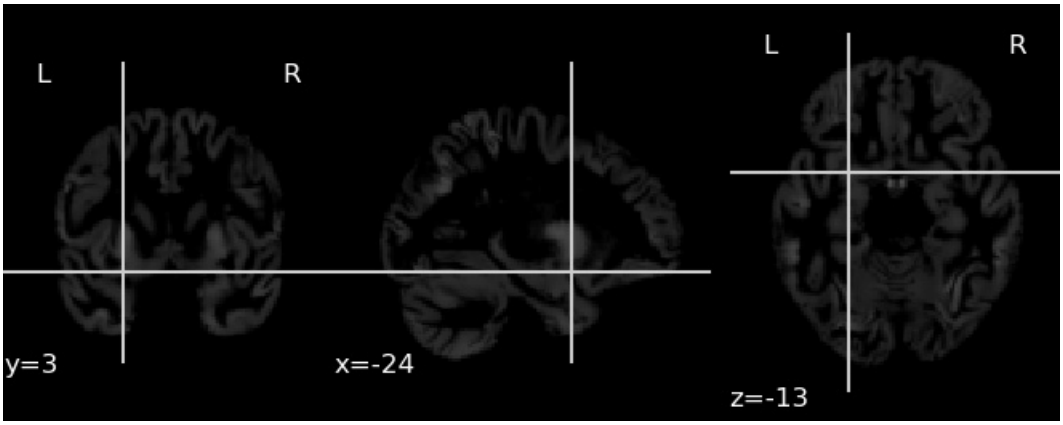



(a) Extracted White Matter for Healthy Control subject

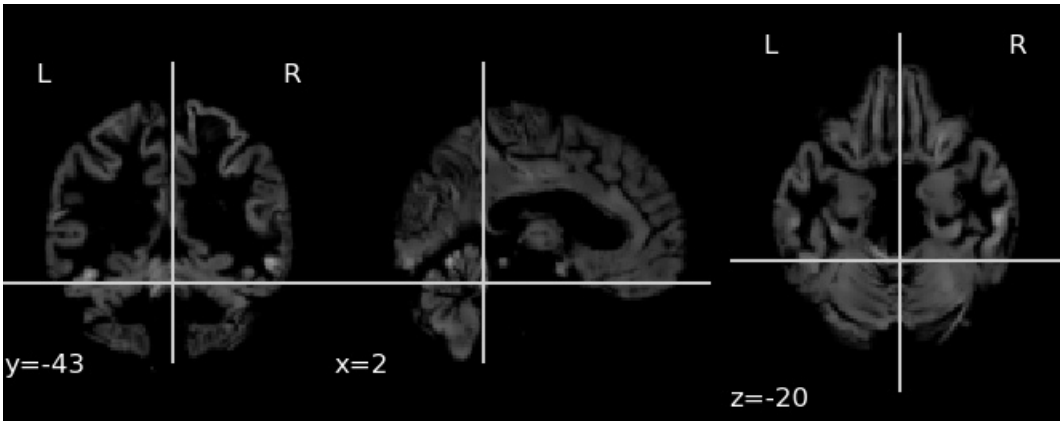



(b) Extracted White Matter for Parkinson's Patient

Figure 3.4: Extracted WM from MRI scans from Figure 3.3




(c) Extracted Gray Matter Healthy Control subject




(d) Extracted Gray Matter for Parkinson's Patient

Figure 3.5: Extracted GM from MRI scans from Figure 3.3

## 3.4 Model Structure

We selected six existing models of the ILSVRC [107] implemented in Pytorch [93] for this experiment.

- ResNet 101 [40]
- SqueezeNet 1.1 [45]
- DenseNet 201 [44]
- VGG 19 [124]
- MobileNet V2 [115]
- ShuffleNet V2 [68]

The six ILSVRC models are available from Torchvision [70] in two versions: without any training (untrained) and trained on the ImageNet dataset. Since the six models were originally designed to process the ImageNet dataset, we had to modify the models in order read the MRI training data. The input layers of all models were changed to accommodate the format of our MRI input and the output layers were changed to predict between 2 classes (PD and HC) instead of the 1000 ImageNet classes. Both untrained and pretrained versions of the models were trained on whole brain, WM and GM scan. Then the models were combined to construct multiple ensemble model blocks and the performances of the resultant architectures were compared to examine if training on the non-PD related ImageNet dataset makes the architectures perform better in PD detection and whether the ensemble architectures outperform individual models. Ensemble architectures often produce better results and our motivation was to compare their performance in this task.

### 3.4.1 Ensemble Architecture - Model 1

In this architecture, we pass our brain scans through six models in parallel and the concatenated output is passed through a Rectified Linear Unit (ReLU) activation function and then to a linear layer with 2 distinct output classes.

The input and output layers are modified to accommodate the shape of the input scans depending whether whole brain scans were used or extracted GM or WM were used. Figure 3.6 shows a visual representation of this architecture.

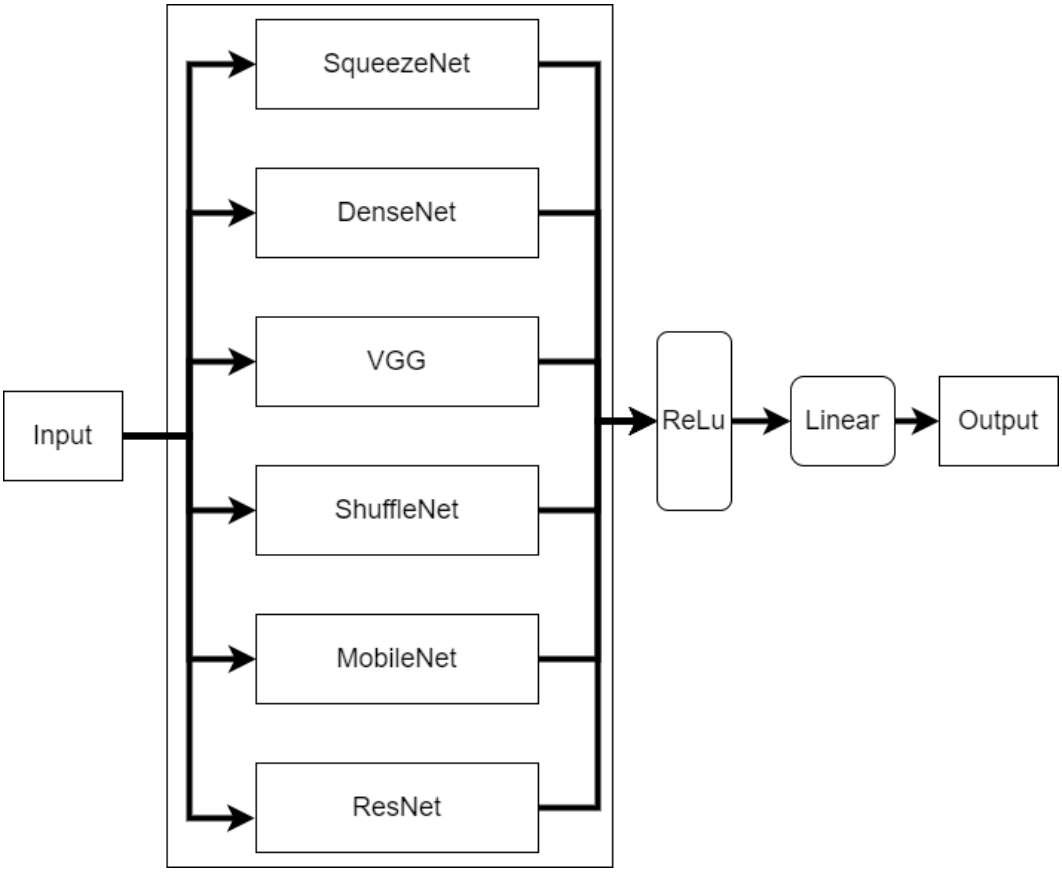



Figure 3.6: Ensemble architecture for whole brain scan: Model 1

### 3.4.2 Ensemble Architecture - Model 2

This architecture is trained on the exclusively extracted GM and WM scans of dimension $121 \times 145 \times 121$. It is comprised of four models. The GM scans are passed through ShuffleNet and SqueezeNet and the WM scans are passed through DenseNet and MobileNet. The output of all models are concatenated and passed through a ReLU activation layer and a linear layer with 2 output classes to get final predictions. Figure 3.7 shows a visual representation of this architecture.

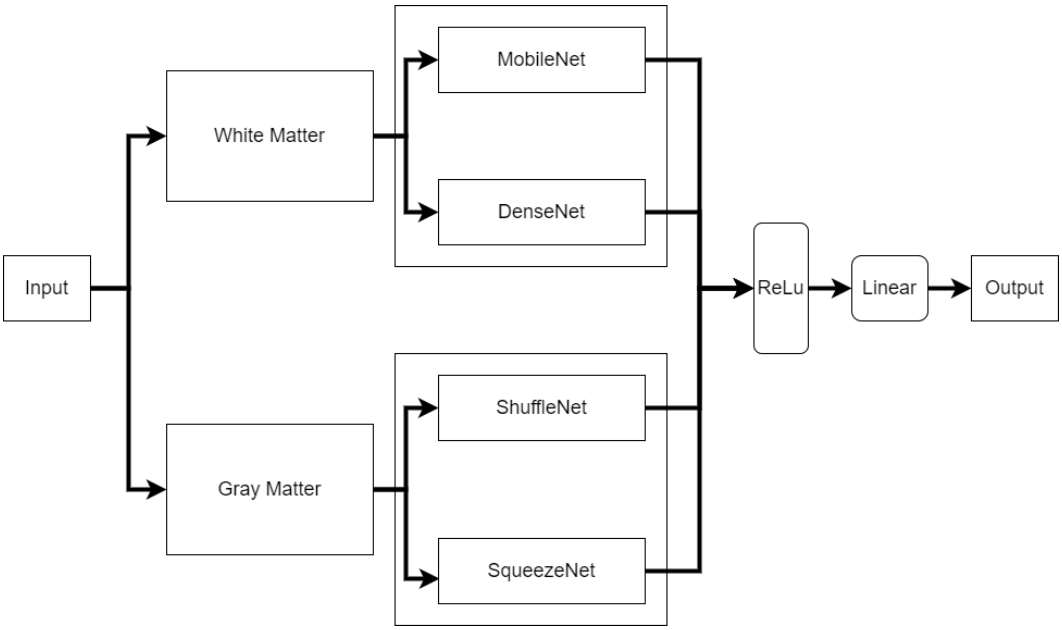



Figure 3.7: Ensemble architecture for Extracted Gray and White Matter Scans : Model 2

### 3.4.3 Ensemble Architecture - Model 3

This architecture was designed with multiple different ensemble structures in mind. The extracted GM and WM scans with dimension $121 \times 145 \times 121$ were passed in parallel through two model blocks, each of which is comprised of multiple ILSVRC models. For our experiment, the models blocks were kept similar. Three instances of the architecture were tested, each with a different sub architecture as both model blocks. The output from both blocks were concatenated and passed through a ReLU activation layer followed by a final linear layer, which predicts between the two output classes. Figure 3.8 shows a visual representation of this architecture.

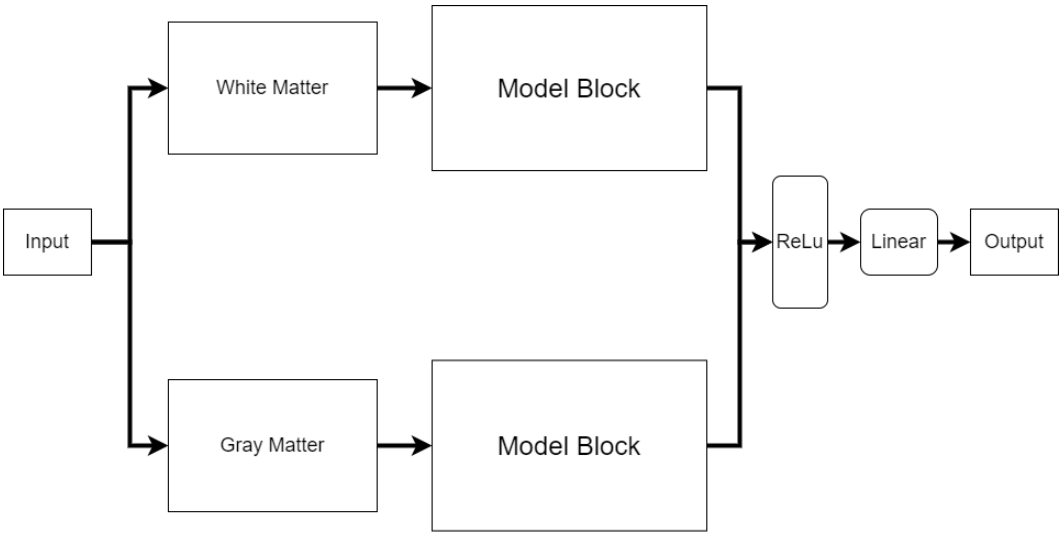



Figure 3.8: Core Architecture

Three distinct model block designs were used for experimentation. Each structure had two versions: pre-trained and untrained with ImageNet data.

#### 3.4.3.1 Sub Architecture 1 - TriNet1

The model block was named TriNet1, due to being comprised of 3 models, DenseNet, ShuffleNet and SqueezeNet in parallel. The input was passed through all three models simultaneously, as shown in Figure 3.9.

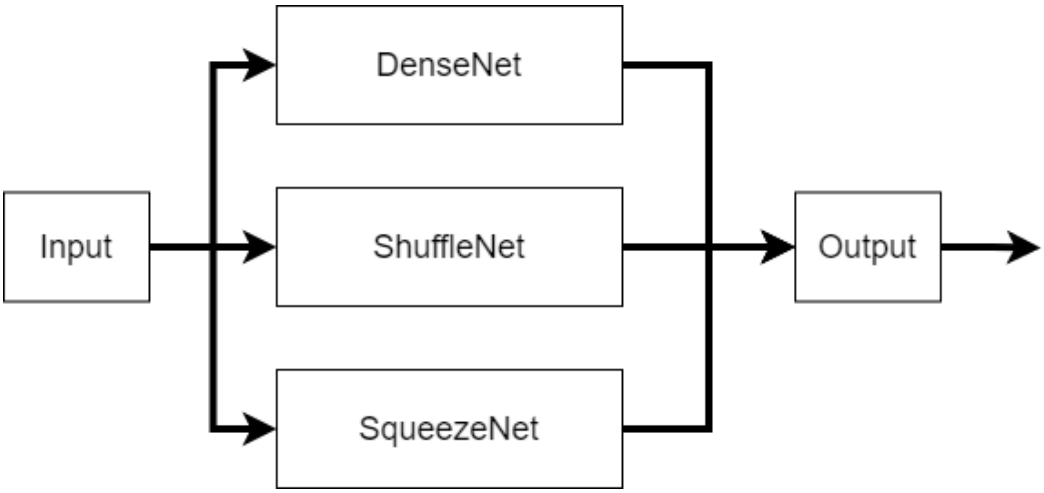



Figure 3.9: Architecture 1 : TriNet1

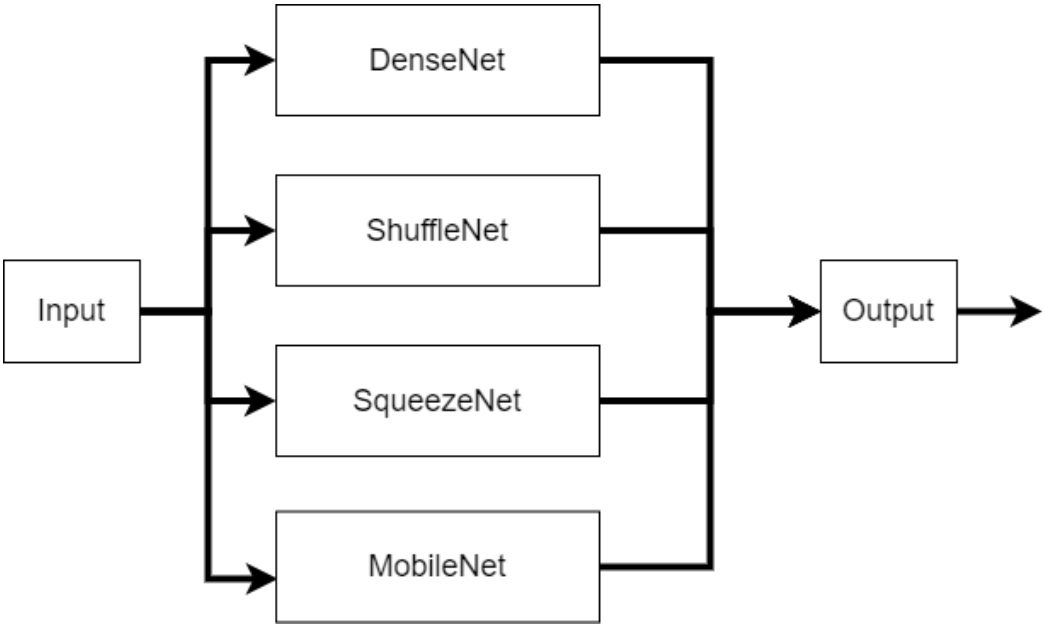



Figure 3.10: Architecture 2 : QuadNet

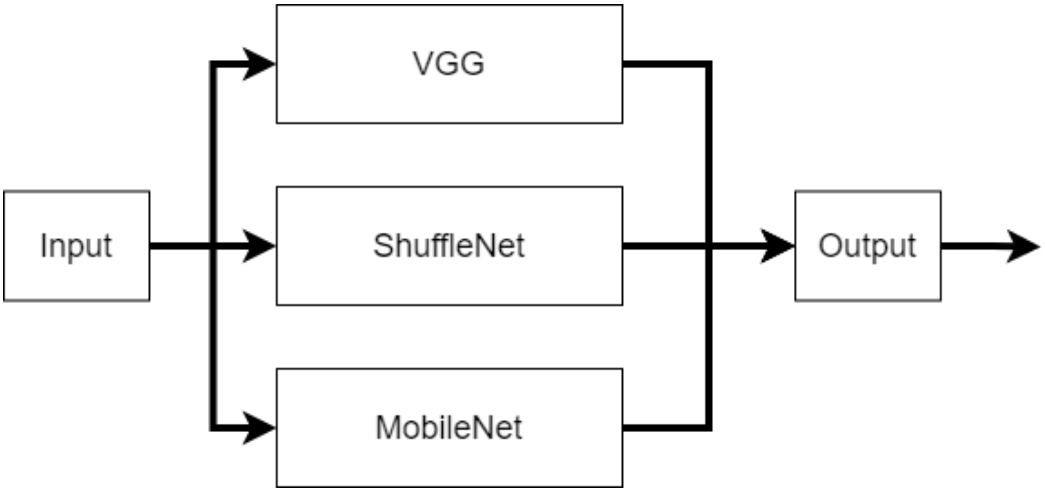



Figure 3.11: Architecture 3 : TriNet2

#### 3.4.3.2 Sub Architecture 2 - QuadNet

This architecture comprised of 4 models and was named QuadNet. The model block was created by adding MobileNet to Block 1, so it was comprised of DenseNet, ShuffleNet, SqueezeNet and MobileNet in parallel. The input was passed through all four models simultaneously, as shown in Figure 3.10.

#### 3.4.3.3 Sub Architecture 3 - TriNet2

This architecture was also comprised of 3 models and was The model block was created with ShuffleNet, VGG and MobileNet in parallel. The input was passed through all three models simultaneously, as shown in Figure 3.11

## 3.5 Experimental Results

Two versions were constructed for each of our ensemble architectures; one with all untrained constituent models and another with all pretrained constituent models. The dataset was divided randomly with evenly split labels, and 80% was selected for training and 20% for testing. Each model was trained for 50 epochs with an Adam Optimizer and Cross Entropy Loss function. At each epoch, the training set was further split randomly, and 20% was selected for validation. We repeated the procedures listed above 5 times to obtain average scores. The models were trained with 3 different learning rates, **.01**, **.001** and **.0001**. After experimenting with different learning rates these 3 produced the best performance. The results are presented in this section. Table 3.2 presents the results of some existing approaches on similar data for reference. This models were trained on some version of the PPMI dataset, but were not class balanced.

Table 3.2: Results of some related works on similar dataset

| Source | Accuracy |
|---|---|
| Focke et al.[27] [GM] | 0.3953 |
| Focke et al.[27] [WM] | 0.4186 |
| Babu et al.[116] [GM] | 0.8721 |
| Rana et al.[99] [GM & WM] | 0.8667 |
| Rana et al.[101] | 0.8967 |

Table 3.3: Results for Resnet with Learning Rate of 0.0001

| Data Type | Pre Trained | Accuracy | MCC | Precision | Recall | F1 Score |
|---|---|---|---|---|---|---|
| Whole Brain Scan | True | 0.650 | 0.306 | 0.655 | 0.650 | 0.650 |
| | False | **0.817** | **0.625** | **0.847** | **0.817** | **0.802** |
| Gray Matter | True | **0.948** | **0.895** | **0.948** | **0.948** | **0.948** |
| | False | 0.522 | 0.064 | 0.751 | 0.522 | 0.363 |
| White Matter | True | **0.963** | **0.925** | **0.955** | **0.955** | **0.955** |
| | False | 0.526 | 0.052 | 0.639 | 0.524 | 0.444 |
| Smooth Gray Matter | True | **0.708** | **0.412** | **0.773** | **0.708** | **0.662** |
| | False | 0.483 | -0.047 | 0.430 | 0.483 | 0.338 |
| Smooth White Matter | True | **0.850** | **0.710** | **0.817** | **0.779** | **0.755** |
| | False | 0.513 | 0.036 | 0.476 | 0.498 | 0.425 |

Table 3.4: Results for VGG with Learning Rate of 0.0001

| Data Type | Pre Trained | Accuracy | MCC | Precision | Recall | F1 Score |
|---|---|---|---|---|---|---|
| Whole Brain Scan | True | 0.775 | 0.561 | 0.785 | 0.775 | **0.773** |
| | False | 0.775 | **0.587** | **0.813** | 0.775 | 0.768 |
| Gray Matter | True | **0.925** | **0.848** | **0.926** | **0.925** | **0.926** |
| | False | 0.545 | 0.139 | 0.584 | 0.545 | 0.535 |
| White Matter | True | **0.940** | **0.880** | **0.933** | **0.933** | **0.933** |
| | False | 0.541 | 0.098 | 0.569 | 0.543 | 0.498 |
| Smooth Gray Matter | True | **0.779** | **0.615** | **0.837** | **0.779** | **0.770** |
| | False | 0.543 | 0.170 | 0.615 | 0.543 | 0.498 |
| Smooth White Matter | True | **0.813** | **0.641** | **0.832** | **0.796** | **0.791** |
| | False | 0.603 | 0.289 | 0.645 | 0.573 | 0.542 |

Table 3.5: Results for DenseNet with Learning Rate of 0.0001

| Data Type | Pre Trained | Accuracy | MCC | Precision | Recall | F1 Score |
|---|---|---|---|---|---|---|
| Whole Brain Scan | True | **0.808** | **0.624** | **0.825** | **0.808** | **0.803** |
| | False | 0.683 | 0.451 | 0.764 | 0.683 | 0.666 |
| Gray Matter | True | **0.918** | **0.838** | **0.921** | **0.918** | **0.918** |
| | False | 0.854 | 0.716 | 0.862 | 0.854 | 0.855 |
| White Matter | True | **0.955** | **0.909** | **0.938** | **0.937** | **0.937** |
| | False | 0.877 | 0.756 | 0.871 | 0.866 | 0.866 |
| Smooth Gray Matter | True | **0.858** | **0.734** | **0.878** | **0.858** | **0.858** |
| | False | 0.667 | 0.337 | 0.670 | 0.667 | 0.666 |
| Smooth White Matter | True | **0.838** | **0.721** | **0.869** | **0.858** | **0.858** |
| | False | 0.700 | 0.410 | 0.701 | 0.684 | 0.673 |

Table 3.6: Results for MobileNet with Learning Rate of 0.0001

| Data Type | Pre Trained | Accuracy | MCC | Precision | Recall | F1 Score |
|---|---|---|---|---|---|---|
| Whole Brain Scan | True | 0.658 | 0.311 | 0.658 | 0.658 | 0.657 |
| | False | **0.683** | **0.454** | **0.766** | **0.683** | **0.667** |
| Gray Matter | True | **0.925** | **0.852** | **0.927** | **0.925** | **0.925** |
| | False | 0.534 | 0.041 | 0.526 | 0.534 | 0.496 |
| White Matter | True | **0.925** | **0.849** | **0.926** | **0.925** | **0.925** |
| | False | 0.604 | 0.203 | 0.565 | 0.569 | 0.550 |
| Smooth Gray Matter | True | **0.779** | **0.587** | **0.808** | **0.779** | **0.774** |
| | False | 0.614 | 0.104 | 0.577 | 0.582 | 0.533 |
| Smooth White Matter | True | **0.850** | **0.690** | **0.829** | **0.815** | **0.812** |
| | False | 0.551 | 0.104 | 0.577 | 0.582 | 0.533 |

Table 3.7: Results for ShuffleNet with Learning Rate of 0.0001

| Data Type | Pre Trained | Accuracy | MCC | Precision | Recall | F1 Score |
|---|---|---|---|---|---|---|
| Whole Brain Scan | True | **0.708** | **0.498** | **0.812** | **0.708** | **0.675** |
| | False | 0.692 | 0.496 | 0.800 | 0.692 | 0.674 |
| Gray Matter | True | **0.918** | **0.836** | **0.920** | **0.918** | **0.918** |
| | False | 0.534 | 0.042 | 0.533 | 0.534 | 0.453 |
| White Matter | True | **0.937** | **0.874** | **0.929** | **0.927** | **0.927** |
| | False | 0.494 | 0.111 | 0.531 | 0.485 | 0.420 |
| Smooth Gray Matter | True | **0.738** | **0.471** | **0.742** | **0.738** | **0.734** |
| | False | 0.476 | -0.035 | 0.473 | 0.476 | 0.392 |
| Smooth White Matter | True | **0.708** | **0.436** | **0.734** | **0.723** | **0.720** |
| | False | 0.494 | 0.111 | 0.531 | 0.485 | 0.420 |

Table 3.8: Results for SqueezeNet with Learning Rate of 0.0001

| Data Type | Pre Trained | Accuracy | MCC | Precision | Recall | F1 Score |
|---|---|---|---|---|---|---|
| Whole Brain Scan | True | **0.733** | **0.450** | **0.741** | **0.733** | **0.722** |
| | False | 0.675 | 0.357 | 0.681 | 0.675 | 0.674 |
| Gray Matter | True | **0.873** | **0.747** | **0.874** | **0.873** | **0.873** |
| | False | 0.757 | 0.542 | 0.790 | 0.757 | 0.748 |
| White Matter | True | **0.948** | **0.898** | **0.912** | **0.910** | **0.910** |
| | False | 0.765 | 0.530 | 0.779 | 0.761 | 0.755 |
| Smooth Gray Matter | True | **0.727** | **0.452** | **0.748** | **0.727** | **0.711** |
| | False | 0.543 | 0.000 | 0.295 | 0.543 | 0.382 |
| Smooth White Matter | True | **0.772** | **0.534** | **0.768** | **0.749** | **0.735** |
| | False | 0.730 | 0.457 | 0.541 | 0.637 | 0.538 |

Table 3.9: Results for Ensemble Model 1 for Whole Brain Scans

| Pre Trained | LR | Accuracy | MCC | Precision | Recall | F1 Score |
|---|---|---|---|---|---|---|
| False | 0.0001 | 0.700 ± 0.000 | 0.396 ± 0.000 | 0.762 ± 0.048 | 0.703 ± 0.024 | 0.688 ± 0.032 |
| | 0.001 | **0.733 ± 0.000** | **0.552 ± 0.000** | **0.823 ± 0.009** | **0.758 ± 0.025** | **0.742 ± 0.025** |
| | 0.01 | 0.550 ± 0.000 | 0.000 ± 0.000 | 0.630 ± 0.232 | 0.658 ± 0.077 | 0.593 ± 0.143 |
| True | 0.0001 | 0.700 ± 0.000 | 0.436 ± 0.000 | 0.719 ± 0.087 | 0.686 ± 0.086 | 0.677 ± 0.092 |
| | 0.001 | 0.708 ± 0.000 | 0.414 ± 0.000 | 0.706 ± 0.025 | 0.706 ± 0.024 | 0.704 ± 0.022 |
| | .01 | **0.742 ± 0.000** | **0.515 ± 0.000** | **0.770 ± 0.019** | **0.725 ± 0.024** | **0.709 ± 0.026** |

Table 3.10: Results for Ensemble Model 2 for Gray Matter and White Matter

| Pre Trained | LR | Accuracy | MCC | Precision | Recall | F1 Score |
|---|---|---|---|---|---|---|
| False | 0.0001 | 0.754 ± 0.000 | 0.562 ± 0.000 | 0.837 ± 0.060 | 0.806 ± 0.079 | 0.799 ± 0.084 |
| | 0.001 | **0.791 ± 0.000** | **0.626 ± 0.000** | **0.844 ± 0.046** | **0.826 ± 0.055** | **0.824 ± 0.056** |
| | 0.01 | 0.549 ± 0.000 | 0.000 ± 0.000 | 0.309 ± 0.009 | 0.556 ± 0.008 | 0.397 ± 0.009 |
| True | 0.0001 | **0.948 ± 0.000** | **0.896 ± 0.000** | **0.941 ± 0.006** | **0.939 ± 0.008** | **0.939 ± 0.008** |
| | 0.001 | 0.937 ± 0.000 | 0.871 ± 0.000 | 0.917 ± 0.022 | 0.915 ± 0.022 | 0.915 ± 0.022 |
| | 0.01 | 0.556 ± 0.000 | 0.000 ± 0.000 | 0.687 ± 0.272 | 0.767 ± 0.157 | 0.714 ± 0.229 |

Table 3.11: Results for Ensemble Model 2 for Smoothed Gray Matter and White Matter

| Pre Trained | LR | Accuracy | MCC | Precision | Recall | F1 Score |
|---|---|---|---|---|---|---|
| False | 0.0001 | 0.566 ± 0.000 | **0.173 ± 0.000** | **0.641 ± 0.026** | **0.625 ± 0.042** | **0.611 ± 0.056** |
| | 0.001 | 0.513 ± 0.000 | 0.000 ± 0.000 | 0.530 ± 0.201 | 0.605 ± 0.094 | 0.544 ± 0.154 |
| | 0.01 | **0.577 ± 0.000** | 0.000 ± 0.000 | 0.309 ± 0.019 | 0.556 ± 0.017 | 0.397 ± 0.020 |
| True | 0.0001 | **0.933 ± 0.000** | **0.870 ± 0.000** | **0.921 ± 0.016** | **0.919 ± 0.014** | **0.919 ± 0.015** |
| | 0.001 | 0.899 ± 0.000 | 0.798 ± 0.000 | 0.838 ± 0.048 | 0.813 ± 0.061 | 0.811 ± 0.063 |
| | 0.01 | 0.562 ± 0.000 | 0.000 ± 0.000 | 0.490 ± 0.247 | 0.645 ± 0.118 | 0.539 ± 0.190 |

Table 3.12: Results for Ensemble Model 3 with Trinet1 for Gray Matter and White Matter

| Pre Trained | LR | Accuracy | MCC | Precision | Recall | F1 Score |
|---|---|---|---|---|---|---|
| False | 0.0001 | **0.882 ± 0.000** | **0.801 ± 0.000** | **0.874 ± 0.070** | **0.837 ± 0.121** | **0.823 ± 0.141** |
| | 0.001 | 0.862 ± 0.000 | 0.725 ± 0.000 | 0.686 ± 0.305 | 0.767 ± 0.186 | 0.712 ± 0.263 |
| | 0.01 | 0.519 ± 0.000 | 0.000 ± 0.000 | 0.270 ± 0.011 | 0.520 ± 0.011 | 0.356 ± 0.012 |
| True | 0.0001 | 0.903 ± 0.000 | 0.811 ± 0.000 | 0.926 ± 0.018 | 0.923 ± 0.019 | 0.923 ± 0.019 |
| | 0.001 | **0.922 ± 0.000** | **0.844 ± 0.000** | **0.944 ± 0.016** | **0.942 ± 0.014** | **0.941 ± 0.014** |
| | 0.01 | 0.511 ± 0.000 | 0.000 ± 0.000 | 0.706 ± 0.314 | 0.789 ± 0.196 | 0.733 ± 0.274 |

Table 3.13: Results for Ensemble Model 3 with Trinet1 for smoothed Gray Matter and White Matter

| Pre Trained | LR | Accuracy | MCC | Precision | Recall | F1 Score |
|---|---|---|---|---|---|---|
| False | 0.0001 | **0.745 ± 0.000** | **0.522 ± 0.000** | **0.722 ± 0.041** | **0.704 ± 0.029** | **0.701 ± 0.026** |
| | 0.001 | 0.551 ± 0.000 | 0.000 ± 0.000 | 0.609 ± 0.219 | 0.682 ± 0.095 | 0.625 ± 0.167 |
| | 0.01 | 0.543 ± 0.000 | 0.000 ± 0.000 | 0.273 ± 0.016 | 0.522 ± 0.016 | 0.358 ± 0.018 |
| True | 0.0001 | 0.858 ± 0.000 | 0.736 ± 0.000 | 0.902 ± 0.022 | 0.894 ± 0.029 | 0.894 ± 0.030 |
| | 0.001 | **0.944 ± 0.000** | **0.889 ± 0.000** | **0.925 ± 0.016** | **0.919 ± 0.018** | **0.918 ± 0.018** |
| | 0.01 | 0.914 ± 0.000 | 0.828 ± 0.000 | 0.863 ± 0.044 | 0.854 ± 0.054 | 0.854 ± 0.054 |

Table 3.14: Results for Ensemble Model 3 with QuadNet for Gray Matter and White Matter

| Pre Trained | LR | Accuracy | MCC | Precision | Recall | F1 Score |
|---|---|---|---|---|---|---|
| False | 0.0001 | **0.963 ± 0.000** | **0.925 ± 0.000** | **0.890 ± 0.101** | **0.848 ± 0.159** | **0.834 ± 0.179** |
| | 0.001 | 0.813 ± 0.000 | 0.651 ± 0.000 | 0.667 ± 0.274 | 0.740 ± 0.151 | 0.684 ± 0.226 |
| | 0.01 | 0.496 ± 0.000 | 0.000 ± 0.000 | 0.269 ± 0.023 | 0.519 ± 0.022 | 0.355 ± 0.025 |
| True | 0.0001 | **0.963 ± 0.000** | **0.927 ± 0.000** | **0.951 ± 0.010** | **0.950 ± 0.009** | **0.950 ± 0.009** |
| | 0.001 | 0.918 ± 0.000 | 0.838 ± 0.000 | 0.934 ± 0.010 | 0.933 ± 0.011 | 0.933 ± 0.011 |
| | 0.01 | 0.944 ± 0.000 | 0.891 ± 0.000 | 0.922 ± 0.018 | 0.919 ± 0.018 | 0.919 ± 0.018 |

Table 3.15: Results for Ensemble Model 3 with QuadNet for Smoothed Gray Matter and White Matter

| Pre Trained | LR | Accuracy | MCC | Precision | Recall | F1 Score |
|---|---|---|---|---|---|---|
| False | 0.0001 | 0.708 ± 0.000 | 0.447 ± 0.000 | **0.722 ± 0.048** | 0.690 ± 0.025 | 0.674 ± 0.014 |
| | 0.001 | **0.738 ± 0.000** | **0.499 ± 0.000** | 0.441 ± 0.229 | 0.598 ± 0.100 | 0.487 ± 0.173 |
| | 0.01 | 0.524 ± 0.000 | 0.000 ± 0.000 | 0.271 ± 0.053 | 0.519 ± 0.034 | 0.356 ± 0.038 |
| True | 0.0001 | 0.906 ± 0.000 | 0.813 ± 0.000 | 0.902 ± 0.003 | 0.898 ± 0.006 | 0.898 ± 0.006 |
| | 0.001 | **0.948 ± 0.000** | **0.899 ± 0.000** | **0.914 ± 0.028** | **0.911 ± 0.027** | **0.911 ± 0.027** |
| | 0.01 | 0.753 ± 0.000 | 0.533 ± 0.000 | 0.625 ± 0.243 | 0.683 ± 0.197 | 0.624 ± 0.180 |

Table 3.16: Results for Ensemble Model 3 with TriNet2 for Gray Matter and White Matter

| Pre Trained | LR | Accuracy | MCC | Precision | Recall | F1 Score |
|---|---|---|---|---|---|---|
| False | 0.0001 | 0.515 ± 0.000 | 0.000 ± 0.000 | **0.487 ± 0.182** | 0.534 ± 0.026 | 0.412 ± 0.049 |
| | 0.001 | **0.649 ± 0.000** | **0.322 ± 0.000** | 0.404 ± 0.189 | **0.563 ± 0.063** | **0.452 ± 0.136** |
| | 0.01 | 0.541 ± 0.000 | 0.000 ± 0.000 | 0.301 ± 0.009 | 0.549 ± 0.008 | 0.389 ± 0.009 |
| True | 0.0001 | 0.955 ± 0.000 | 0.910 ± 0.000 | 0.947 ± 0.030 | 0.947 ± 0.030 | 0.947 ± 0.030 |
| | 0.001 | **0.966 ± 0.000** | **0.932 ± 0.000** | **0.955 ± 0.020** | **0.954 ± 0.020** | **0.954 ± 0.020** |
| | 0.01 | 0.866 ± 0.000 | 0.739 ± 0.000 | 0.891 ± 0.009 | 0.886 ± 0.014 | 0.886 ± 0.016 |

Table 3.17: Results for Ensemble Model 3 with TriNet2 for smoothed Gray Matter and White Matter

| Pre Trained | LR | Accuracy | MCC | Precision | Recall | F1 Score |
|---|---|---|---|---|---|---|
| False | 0.0001 | **0.599 ± 0.000** | **0.175 ± 0.000** | **0.614 ± 0.055** | **0.615 ± 0.055** | **0.611 ± 0.056** |
| | 0.001 | 0.536 ± 0.000 | 0.000 ± 0.000 | 0.500 ± 0.154 | 0.583 ± 0.048 | 0.528 ± 0.114 |
| | 0.01 | 0.539 ± 0.000 | 0.000 ± 0.000 | 0.288 ± 0.010 | 0.537 ± 0.009 | 0.375 ± 0.011 |
| True | 0.0001 | **0.951 ± 0.000** | **0.902 ± 0.000** | **0.944 ± 0.009** | **0.943 ± 0.010** | **0.943 ± 0.010** |
| | 0.001 | 0.903 ± 0.000 | 0.806 ± 0.000 | 0.888 ± 0.044 | 0.886 ± 0.046 | 0.887 ± 0.046 |
| | 0.01 | 0.547 ± 0.000 | 0.000 ± 0.000 | 0.453 ± 0.224 | 0.613 ± 0.099 | 0.500 ± 0.167 |

Table 3.18: Results for Ensemble Model 1 with Pre trained constituent models for Gray Matter and White Matter

| Datatype | LR | Accuracy | MCC | Precision | Recall | F1 Score |
|---|---|---|---|---|---|---|
| WM | 0.0001 | **0.963 ± 0.000** | **0.928 ± 0.000** | **0.944 ± 0.028** | **0.943 ± 0.028** | **0.943 ± 0.028** |
| | 0.001 | 0.907 ± 0.000 | 0.815 ± 0.000 | 0.936 ± 0.021 | 0.934 ± 0.023 | 0.934 ± 0.023 |
| | 0.01 | 0.933 ± 0.000 | 0.866 ± 0.000 | 0.698 ± 0.314 | 0.781 ± 0.196 | 0.726 ± 0.275 |
| GM | 0.0001 | **0.948 ± 0.000** | **0.896 ± 0.000** | **0.936 ± 0.013** | **0.935 ± 0.013** | **0.935 ± 0.013** |
| | 0.001 | 0.918 ± 0.000 | 0.835 ± 0.000 | 0.916 ± 0.013 | 0.915 ± 0.021 | 0.915 ± 0.021 |
| | 0.01 | 0.892 ± 0.000 | 0.784 ± 0.000 | 0.691 ± 0.301 | 0.774 ± 0.183 | 0.719 ± 0.261 |

## 3.6 Discussion

The tables report multiple evaluation metric scores including Accuracy, Precision, Recall, $F_1$ score, MCC score. The best measure for understanding whether a model is performing well in all 4 criteria of a standard 2 x 2 confu-

sion matrix is the MCC score. All scores are reported in the range of (0, 1), except MCC score, which is in the range of (-1, 1). The scores are reported in $Mean \pm Standard\ Deviation$ format. The best scores for each model using the same modality of data but with different window sizes were reported in bold font.

Table 3.3 presents the performance of Resnet [40] architecture with modified input and output layers and with a 0.0001 learning rate with all our data modalities. Comparing the overall performance, we can conclude that WM achieves the best performance across all metrics for PD detection. GM performed highly across all metrics as well, although the scores were lower than that of WM. In fact, modified Resnet with WM achieved one of the best scores across all metrics among all of the tested models. Using smoothed scans did not improve the performance and both smoothed WM and GM had lower scores than original scans. Whole Brain scans had better performance than smoothed GM but worse scores than smoothed WM. While testing on both original and smoothed GM and WM, models pretrained with Imagenet data showed significantly better performances. However, for whole brain scans, models without any previous training achieved better results.

The scores of VGG [124] architecture with modified input and output layers and with a 0.0001 learning rate with all of the listed data modalities are presented in Table 3.4. WM achieved the best performance across all metrics for PD detection for VGG. GM also achieved high scores across all metrics, but the scores were slightly lower than that of WM. Both smoothed WM and GM had lower scores. Whole Brain scans had the worst performance among all the data modalities. Pretrained models achieved better metric scores across all modalities except with whole brain scans, which had better scores for some of the metrics using untrained models.

Table 3.5 present the results for Densenet [44]. The input and output layers of the Densenet architecture were modified to accommodate our data. The learning rate was fixed at 0.0001. Pretrained models on ImageNet data achieved better metric scores across all modalities for this model. WM produced the best performance, followed by GM, smoothed WM, smoothed GM

and finally whole brain scans.

The results for modified MobileNet [115] architecture with a learning rate of 0.0001 are presented in Table 3.6. Non trained models achieved superior scores for whole brain scans for this model, pretrained models on achieved better scores across all other modalities. GM and WM achieved comparable accuracy with this model, followed by smoothed WM and then smoothed GM. Whole brain scans produced the overall lowest scores for this model.

Table 3.7 lists the metric scores for modified ShuffleNet [68] architecture with a constant learning rate of 0.0001. It can be seen that pretrained models on the ImageNet dataset outperform non trained models with all data modalities . WM provides the best performance, closely followed by GM. Smoothed GM perform worse than GM but better than whole brain scans and smoothed WM. Whole brain scans also outperform smoothed WM.

The performance of modified SqueezeNet [45] was listed in Table 3.8. The learning rate was once again kept constant at 0.0001. Pretrained models on ImageNet data outperform non trained versions across all modalities for this model. WM achieved significantly better scores, followed by GM. The scores for whole brain scan, smoothed GM and smoothed WM were more or less similar.

Table 3.9 presents the results for Ensemble Architecture 1 defined in Section 3.4.1 trained on whole brain scans with three different learning rates. We see that for learning rates of 0.0001 and 0.001 pretrained models do not significantly impact the performance. In fact the highest scores were achieved a learning rate of 0.001 with non trained model. However, for a learning rate of 0.01, pretrained models significantly outperform ensemble model 1 without any previous training.

The results for Ensemble Architecture 2 defined in Section 3.4.2 trained on GM and WM scans with three different learning rates are shown in Table 3.10. Pretrained models show significantly higher metric scores compared to non trained models for this architecture. Table 3.11 presents the scores for the same model, but with smoothed GM and WM scans. Pretrained models outperform nontrained models with smoothed scans as well, however for simi-

lar parameters, the performance of using smoothed GM and smoothed WM is slightly worse than using non smoothed GM and WM. For both non smooth and smooth scans, a learning rate of 0.0001 generated the best results.

Table 3.12 presents the results for Ensemble Architecture 3 [3.4.3] with TriNet1 defined in Section 3.4.3.1. For non trained constituent models, the best scores were achieved with a learning rate of 0.0001, whereas a learning rate of 0.001 provided best results for pretrained constituent models. Using pretrained models to construct the ensemble architecture produced better results when keeping all other parameters fixed. The scores of same model with smoothed GM and WM are listed in Table 3.13. For this model, using non trained models to construct the ensemble architecture produces lower metric scores when compared to previous models where GM and WM were used without smoothing. However keeping all parameters fixed and using pretrained constituent models gives better results with learning rates of 0.001 and 0.01. The performance for the learning rate of 0.0001 is slightly worse when using smoothed GM and WM.

The results for Ensemble Architecture 3 [3.4.3] with QuadNet defined in Section 3.4.3.2 is listed in Table 3.14. For both non trained and pretrained constituent models, the best scores were achieved with a learning rate of 0.0001. The scores for both approaches were pretty close with 0.0001 learning rate, but for other learning rates of 0.001 and 0.01 pretrained models to construct the ensemble architecture produced significantly better results keeping all other parameters fixed. The scores of the same QuadNet model with smoothed GM and WM are shown in Table 3.15. For QuadNet, non trained models produce lower metric scores when compared to previous models where GM and WM were used without smoothing. Using pretrained constituent models gives significantly better results with all learning rates keeping all parameters fixed. Overall, the performance is better when using non smoothed GM and WM.

Table 3.16 contains the results for Ensemble Architecture 3 [3.4.3] with TriNet2 defined in Section 3.4.3.3. For both non trained and pretrained constituent models, the best scores were achieved with a learning rate of 0.001.

The scores for pretrained models to construct the TriNet2 architecture produced significantly better results keeping all other parameters fixed. The scores for the TriNet2 with smoothed GM and WM are shown in Table 3.17. For TriNet2, non trained models produce similar metric scores with smoothed and non smoothed GM and WM. Using pretrained constituent models gives significantly better results with all learning rates keeping all parameters fixed. When using smoothed scans, the best results were obtained with a learning rate of 0.0001. Overall, the performance is better when using non smoothed GM and WM.

Table 3.18 presents the metric scores for Ensemble Model 1 defined in Section 3.4.1 with only non smooth GM and WM. Based on our previous observations it was derived that individually GM and WM would be sufficient in our task of PD detection. Only pretrained constituent models were used as pretrained models have produced superior results in previous experiments. The experiments were done with a total of 3 different learning rates, with 0.0001 producing the best results for both GM and WM. For learning rates of 0.0001 and 0.01, WM produces better results compared to GM, however the results are more close for a learning rate of 0.001.
Overall,

- it can be seen that our proposed architectures perform better than existing models on similar data and achieve above 90% accuracy,
- the scores increase significantly in models pre-trained on ImageNet when using WM and GM scans, however for whole brain scans, models without any previous training produced better scores.
- it was also noted that extracted GM and WM produce better performance compared to whole brain scans.
- we also observe that the results using non-smoothed scans are better than using smoothed scans. We believe that fine representative details are likely removed during the smoothing process.

- it was concluded that for detection we did not need both GM and WM. Individually both of them produced above 90% detection accuracy, but overall WM produces better performance.

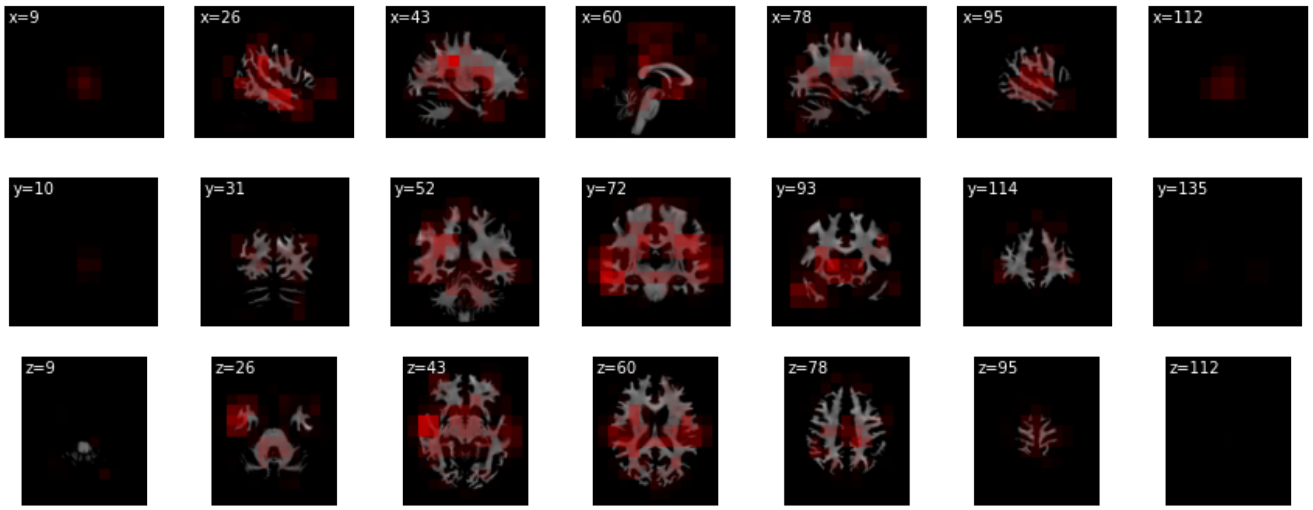
## 3.7 Occlusion Analysis

To understand which regions of the brain are important in the decision making process, we performed a slightly modified version of occlusion analysis proposed by Rieke et al. [104] to fit our data. In this analysis, usually a part of the scan is occluded with gray or white patch and the output from the network is recalculated. The occluded region is considered to be important if the probability of the target class decreases compared to the original image. The heatmap of relevance is calculated by sliding the patch across the image and plotting the difference in the probability in red. The brightness of the shade of red indicates the importance of the region. The relevance was calculated such that the sum of relevance of all areas was 1. The heatmaps presented in the following sections contain slices taken from the original MRI scan at specific $x$,$y$ or $z$ coordinates and overlaying the difference in probability for that point. Occlusion analysis was performed for three of the models with parameters that produced the best performance. The models were modified Resnet [40], Ensemble Architecture Model 1[3.4.1] and Ensemble Architecture Model 2[3.4.2]. All models were pretrained with ImageNet [20] dataset, and they were trained with a learning rate of 0.0001. In our experiments, GM and WM produced superior results than whole brain scans, so only models trained on GM and WM were selected for occlusion analysis. To better understand the heatmaps we also calculated the relevance per brain area using methods provided by Rieke et al [104].

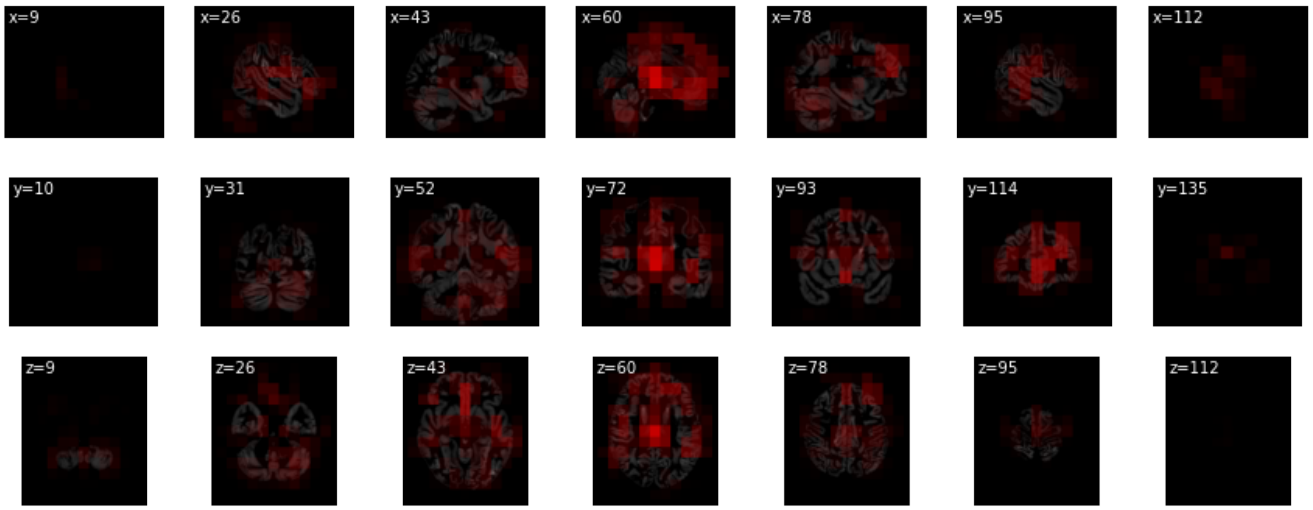
### 3.7.1 Occlusion Analysis for Modified ResNet

Modified Resnet [40] produced the best results out of our individual models 3.3. Two versions of the model were trained, one on WM and the other on GM. Figure 3.12 presents the relevance heatmap for WM and GM using pre-

trained ResNet [40]. Figure 3.13 shows that the Middle Temporal Gyrus and Superior Temporal Gyrus were significant in the decision making process for the model when using WM, followed by the Postcentral Gyrus region. The rest of the regions were of relatively low relevance. Figure 3.14 shows that when using GM, the relevance were comparatively more evenly distributed. Middle Temporal Gyrus was once again vital in the decision making process, followed closely by Middle Frontal Gyrus, Frontal Superior Medial Gyrus, Thalamus and Superior Temporal Gyrus.




(a) Heatmap for White Matter




(b) Heatmap for Gray Matter

Figure 3.12: Relevance Heatmaps for Occlusion of Gray Matter and White Matter images using Pretrained ResNet [40]

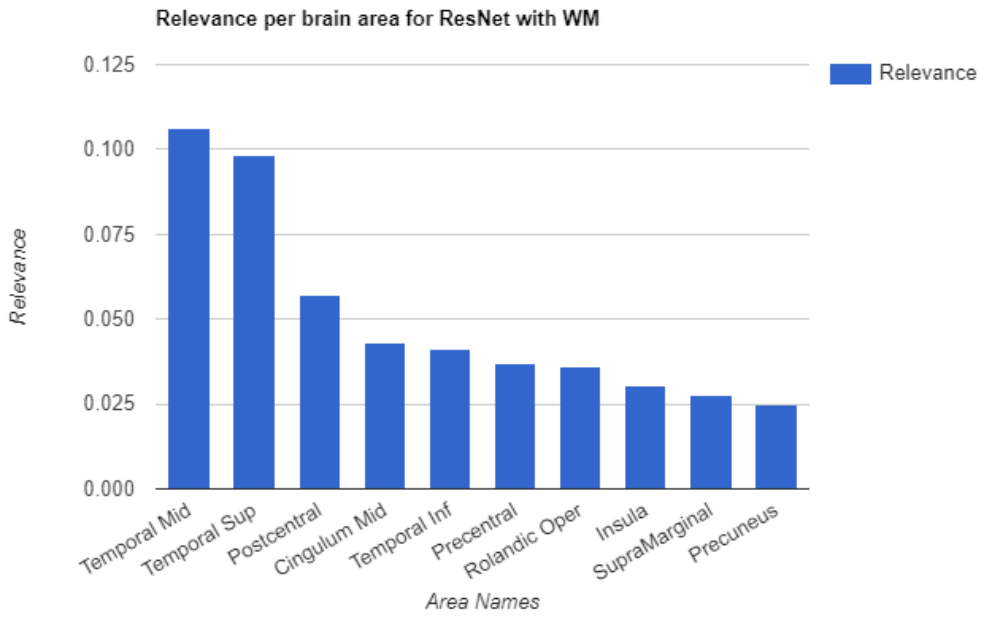



Figure 3.13: Relevance per brain area for White Matter for Pretrained ResNet [40]

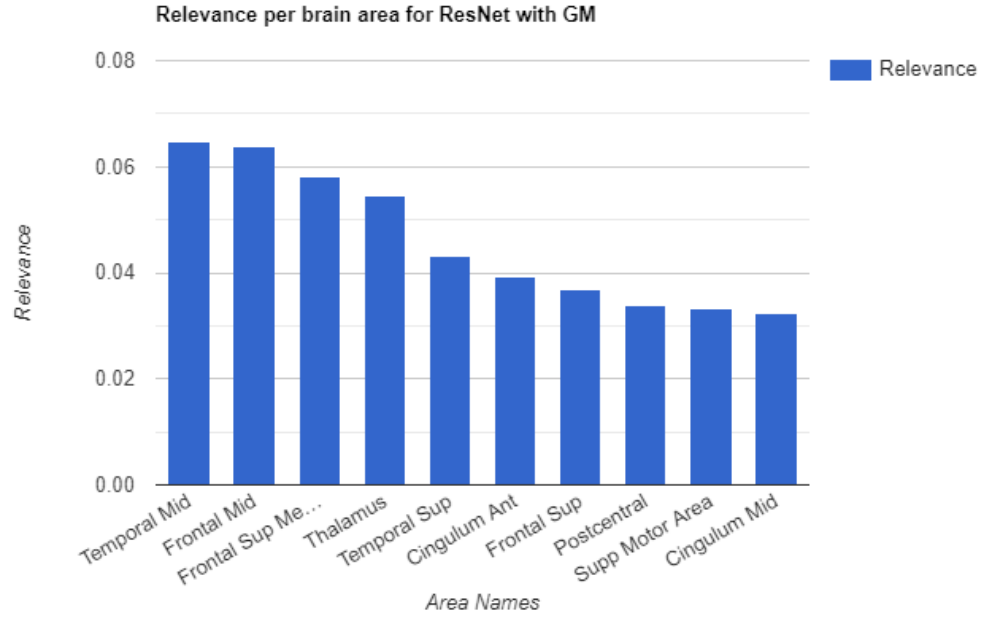
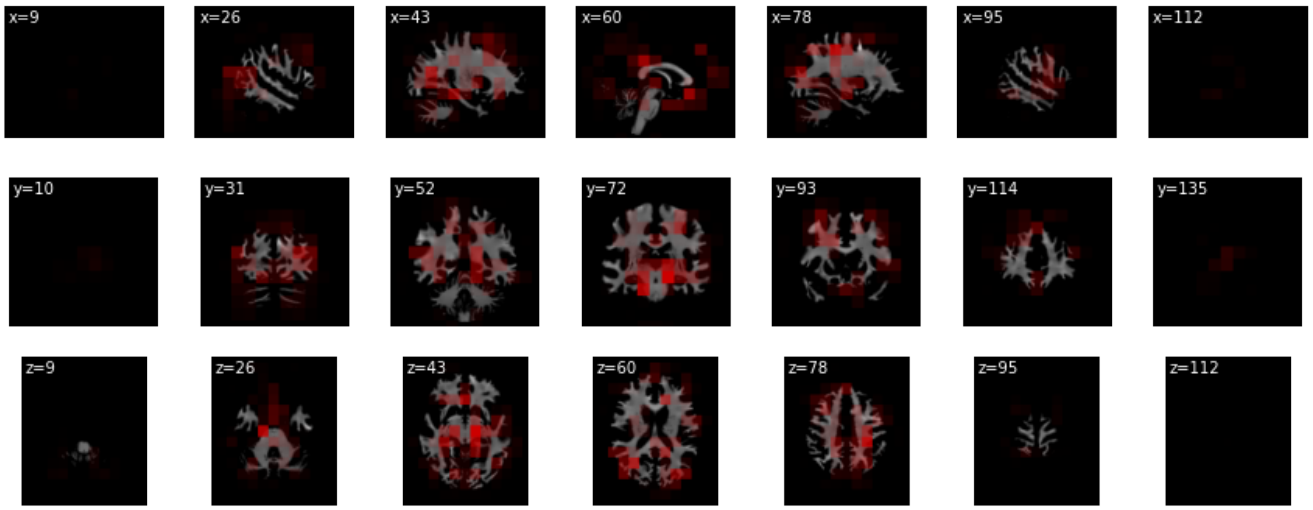


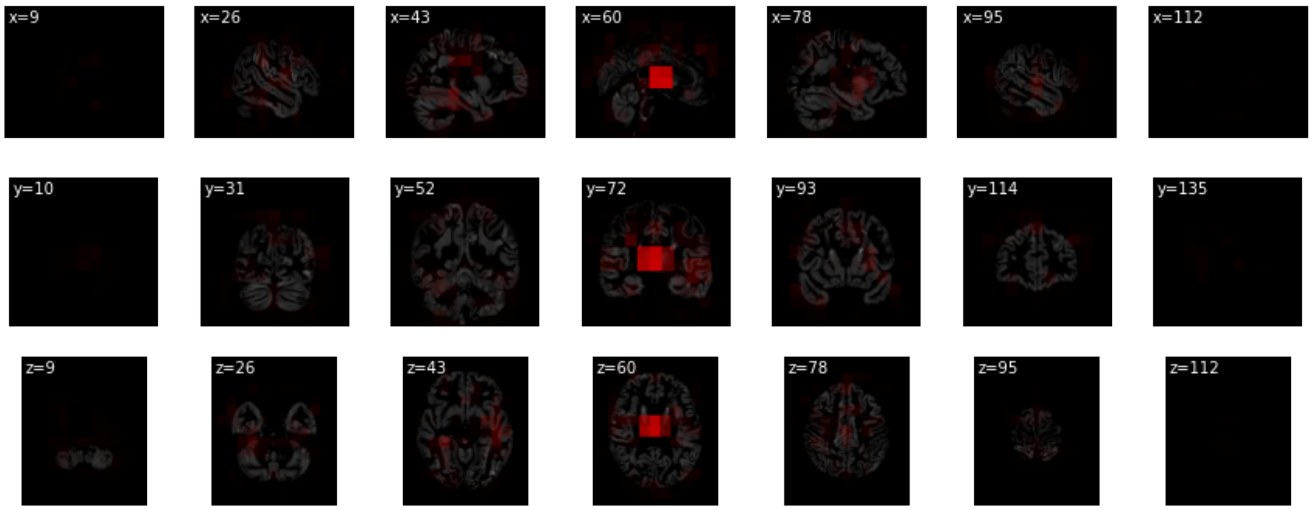

Figure 3.14: Relevance per brain area for Gray Matter for Pretrained ResNet [40]

### 3.7.2 Occlusion Analysis for Ensemble Architecture - Model 1

Ensemble Architecture 1[3.4.1] performed very well when trained on GM and WM 3.18. The relevance per brain area was computed on 2 version of the model trained on WM and GM separately. The relevance heatmaps for both types of data are presented in Figure 3.15. Figure 3.16 presents the bar graph for relevance per brain area while using only WM, showing the Middle Frontal, Middle Occipital and Middle Temporal Gyrus to be the three most relevant areas for decision making. However, the Thalamus, Superior Temporal and Middle Temporal Gyrus appear to be the most relevant when using GM as shown in Figure 3.17.




(a) Heatmap for White Matter




(b) Heatmap for Gray Matter

Figure 3.15: Relevance Heatmaps for Occlusion of White Matter and Gray Matter images using Ensemble Architecture 1 with pretrained constituent models

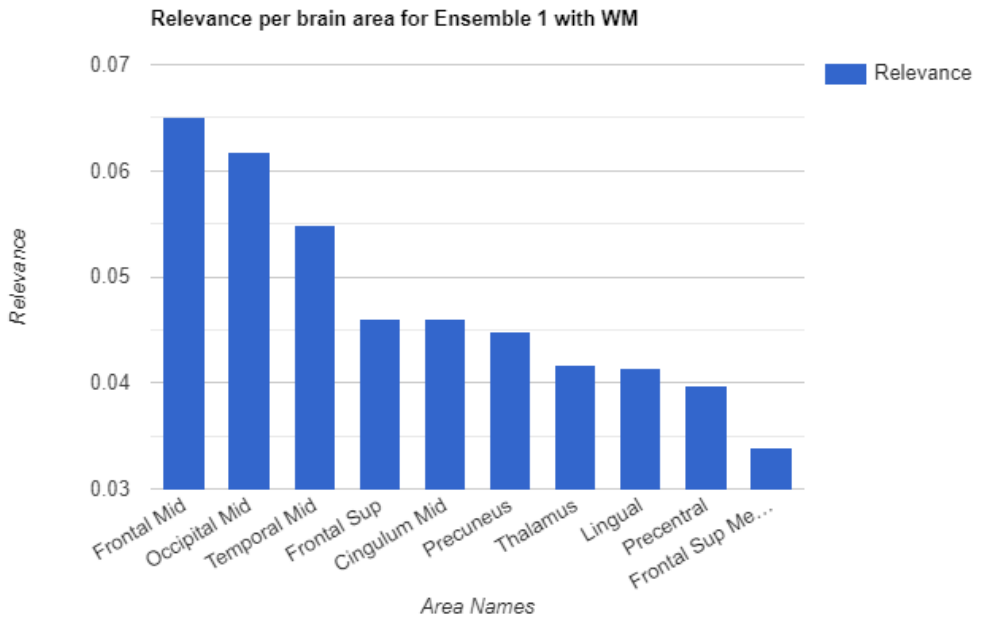



Figure 3.16: Relevance per brain area for White Matter using Ensemble Architecture 1 with pretrained constituent models

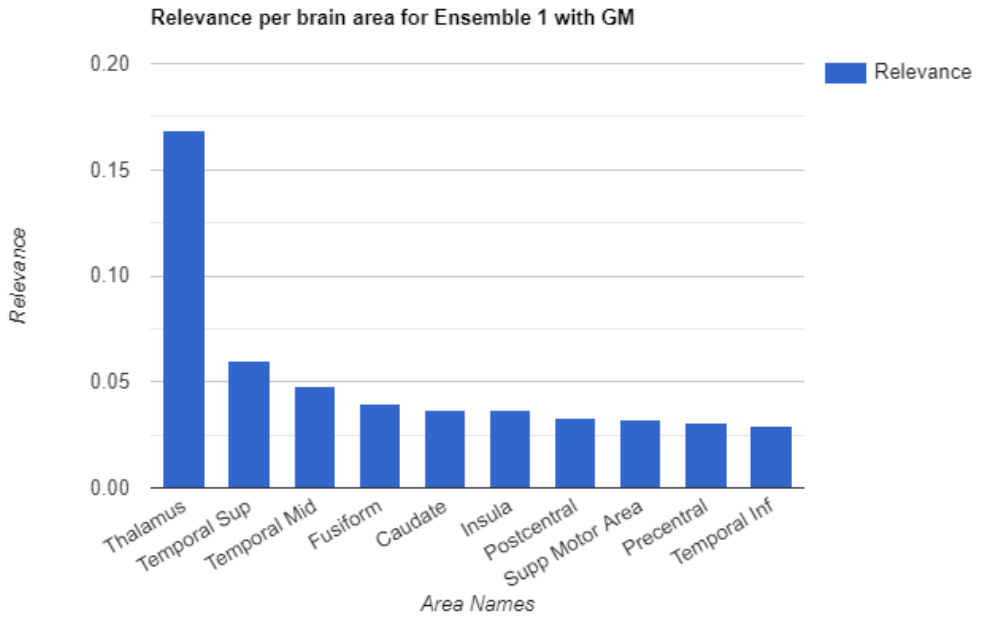
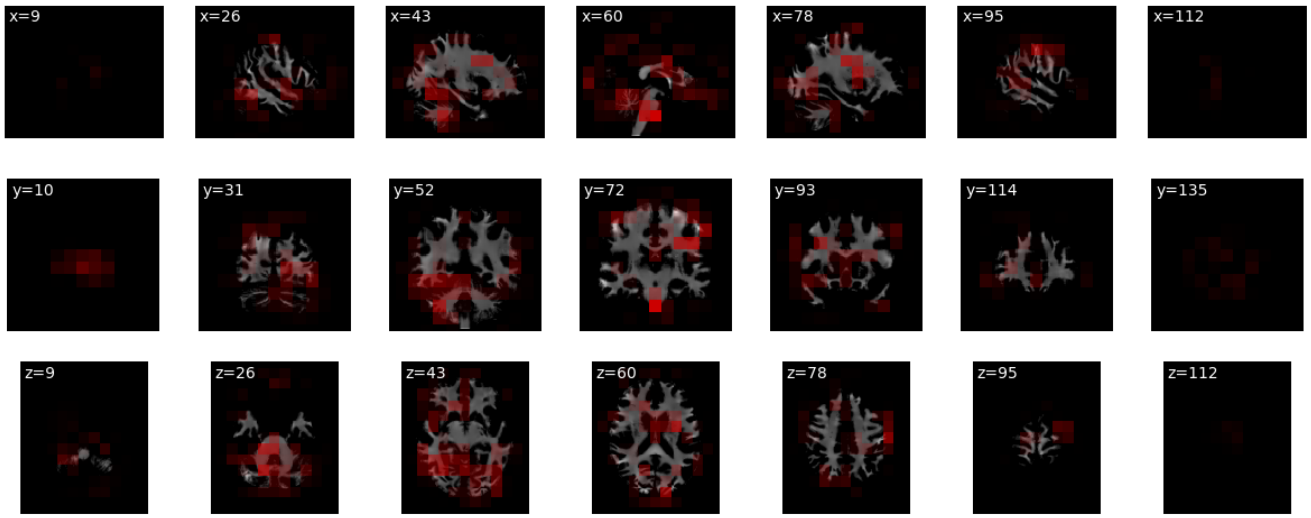


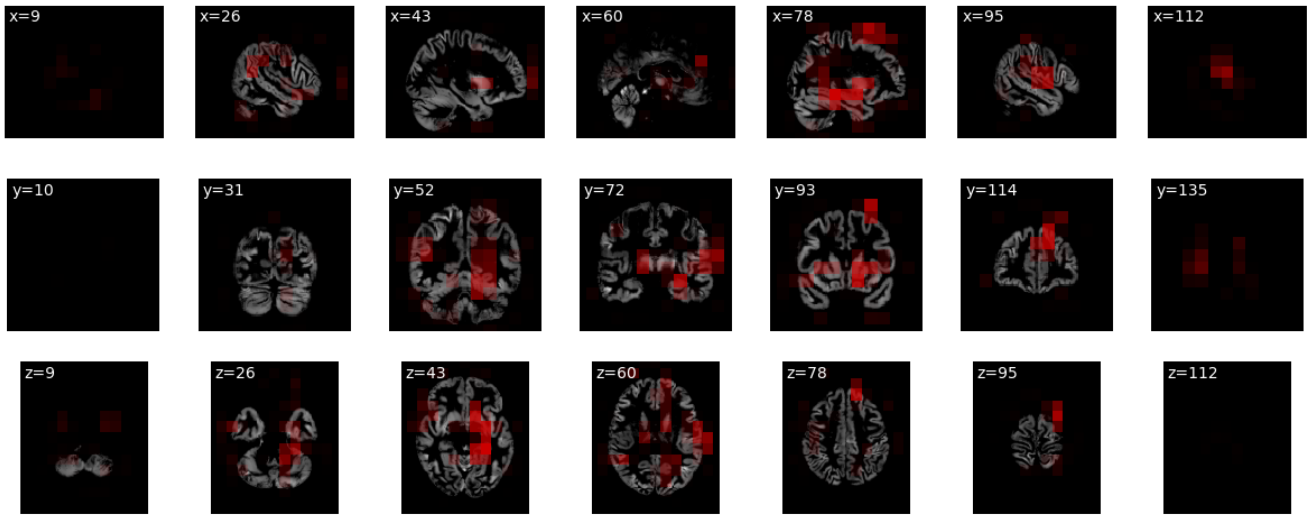

Figure 3.17: Relevance per brain area for Gray Matter using Ensemble Architecture 1 with pretrained constituent models

### 3.7.3 Occlusion Analysis for Ensemble Architecture - Model 2

This architecture[ 3.4.2] was unique from the other architectures we performed occlusion analysis on, in the sense that this architecture needed both GM and WM as input simultaneously. To overcome this issue, the occlusion procedure was repeated twice, once for each of the GM and WM images and then the results were overlayed on top of each other to produce the heatmaps. The resultant heatmaps are presented in Figure 3.18. The relevance per brain area is presented in Figure 3.19 and Figure 3.20. We can see for GM the most focused on area is Superior Frontal, Superior Temporal and Frontal Superior Medial Gyrus and for WM it is Postcentral, Middle Temporal and Fusiform Gyrus.




(a) Heatmap for White Matter




(b) Heatmap for Gray Matter

Figure 3.18: Relevance Heatmaps for Occlusion of White Matter and Gray Matter images using Ensemble Architecture 2 with pretrained constituent models

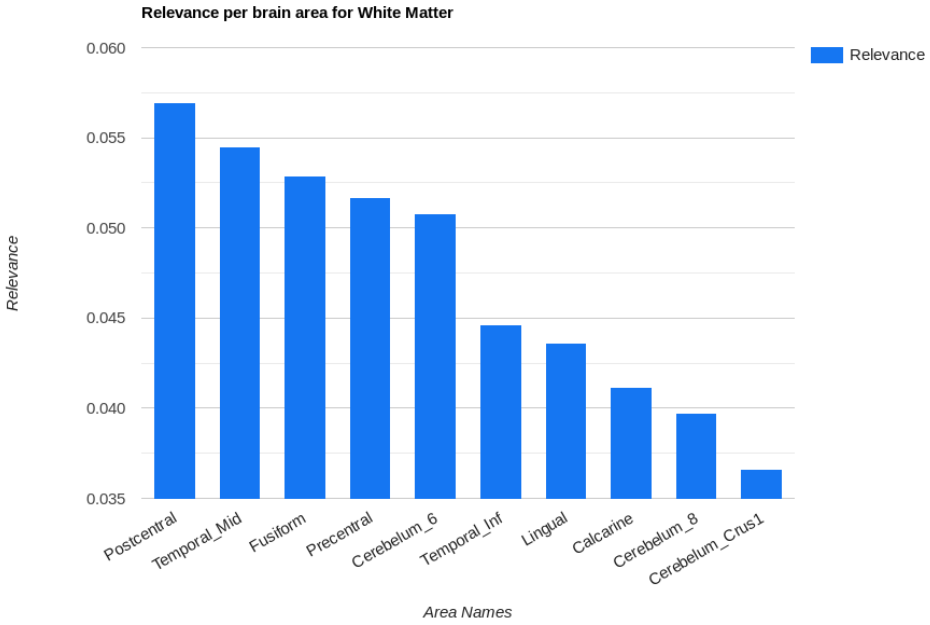



Figure 3.19: Relevance per brain area for White Matter using Ensemble Architecture 2 with pretrained constituent models

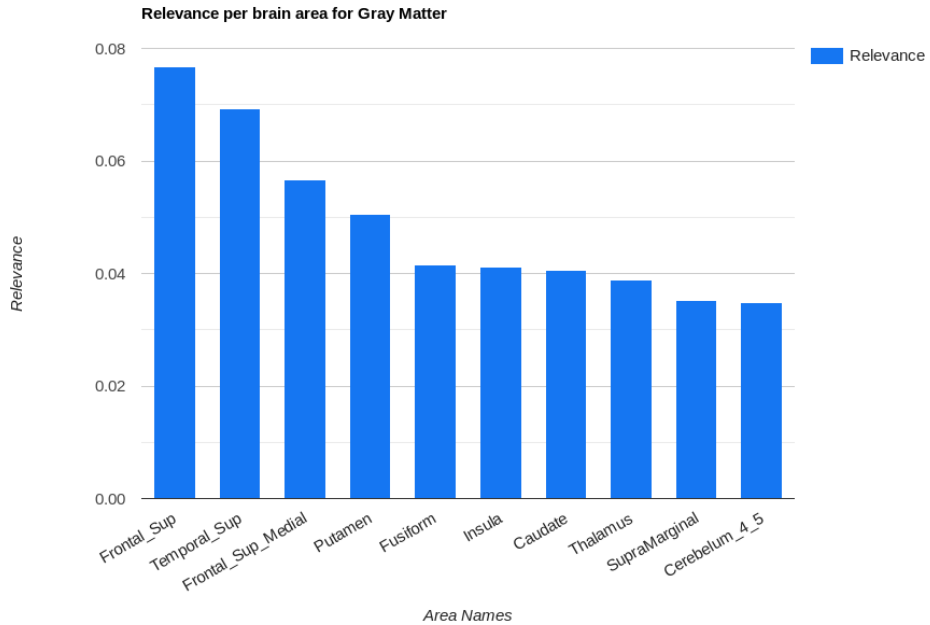



Figure 3.20: Relevance per brain area for Gray Matter using Ensemble Architecture 2 with pretrained constituent models

Overall, we can conclude that,

- Although the relevance of the areas vary from model to model and it is an approximate estimate, some common areas appear to show high importance in all models.
- Superior Temporal Gyrus and Middle Temporal Gyrus are found to be the most common area that models with high performance focus on.
- Postcentral Gyrus, Thalamus and Superior Frontal Gyrus are focused on by most models and appear to have moderately high relevancy in the decision making process.
- The other areas show varying relevance depending the model and the data modality being used.

## 3.8 Conclusion

In this part of the thesis, the author applied transfer learning based approach by applying models trained on the ImageNet [20] dataset on MRI images to detect PD. Pretrained models produced superior metric scores which show a promising direction for research in situations where there is insufficient training data. Multiple novel deep learning architectures for PD detection using Ensemble Learning were proposed, which outperforms related works on similar dataset. It was concluded that pretrained models outperform non trained models for this task. Also, WM by itself produced the best metric scores. We also applied occlusion analysis to identify the region of significance for model decision making process. In future, we want to expand on our research and focus further on the regions that were identified to be most important on the decision making process.

# Chapter 4

# Freezing Of Gait Detection and Prediction

## 4.1 Introduction

In this chapter, we developed DL based techniques and used some of the most widely used time frequency representation techniques as feature set to classify as well as predict the FOG events using data captured from a tri-axial accelerometer sensor. In order to solve the issue of detection latency, we predict the changes in gait before the start of a FOG event. If the onset of FOG events can be accurately predicted, RAS can be applied even before it starts. We use a BiDirectional LSTM architecture with raw signals and handcrafted features. We explored a CNN architecture with multiple visual representation methods including RP, STFT, DWT and PWVD. Experimental results show that our approach give higher accuracy compared with existing state-of-the-art models based on tri-axial accelerometer sensor signals. The performance of each DL model was evaluated with different feature sets and multiple metrics in order to determine the optimal combination of models without bias. Finally, we are able to propose three ensemble architectures, each of which is composed of a selected set of models and features. The ensemble architectures significantly improve the performance of individual models.

Our research group obtained gait data from a multitude of sensors in collaboration with A. T. Still University. In the later parts of this chapter, I propose a system to use proposed models to detect gait from the collected

data using an ensemble architecture.

## 4.2 Data

The publicly available DAPHNet [11] dataset was used for our experiments. The dataset contained data collected from ten PD patients, with seven male and three female experiencing regular FOG in their day to day activities. The average age of the participants was $66.4 \pm 4.8$ years, with an average disease duration of $13.7 \pm 9.67$ years. The average Hoehn and Yahr score was $2.6 \pm 0.65$, indicating that the subjects had mild symptoms with mild balance impairment to moderate balance impairment [42]. Two tri-axial (3D) accelerometer sensors were attached to one of the patient's legs: One was located at the shank just above the ankle, and another was attached to the thigh slightly above the knee. The third sensor was placed at the lower back of the patient. The locations of the sensors are shown in [11] as shown in Figure 4.1.

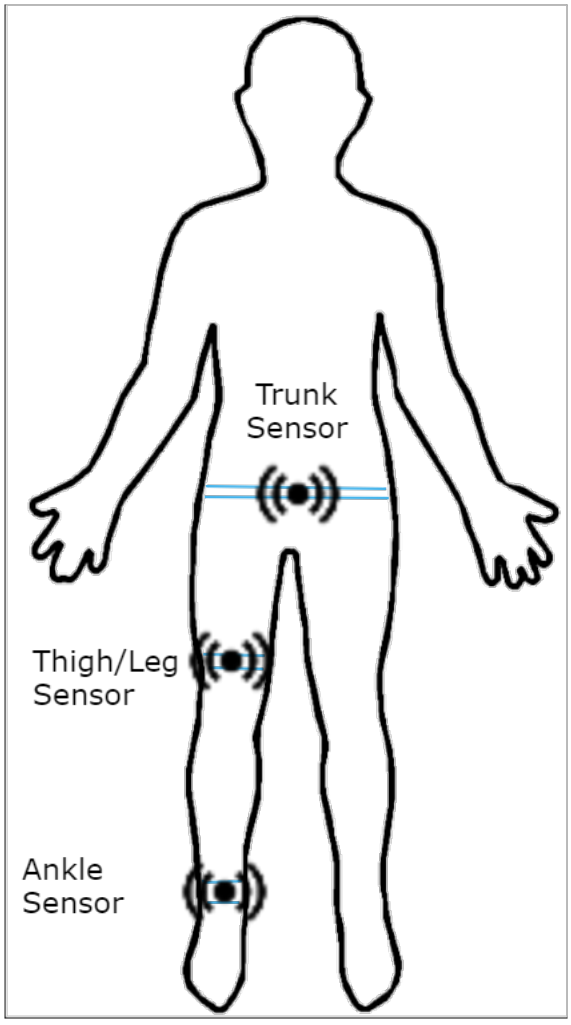



Figure 4.1: Sensor Placement for Data Collection

The dataset contained 237 FOG events which were identified by professional physiotherapists in a post hoc video analysis. Synchronized video recordings were used by physiotherapists to identify the FOG events. The signal point, where the left-right steps alternating, is defined as the start of a FOG event. The point, where pattern resumed is defined as the end of the FOG event. Eight out of the ten subjects experienced FOG during the study, with

the duration of the events ranging from 0.5 seconds to 40.5 seconds. The mean duration was $7.3 \pm 6.7$ seconds. 50% of the FOG episodes were shorter than 5.4 seconds and 93.2% were shorter than 20 seconds. The signals were annotated in three categories:

- 0 - Not part of the experiment; user performed activities are unrelated to the experimental protocol while the sensors were installed.
- 1 - Experiment; no FOG.
- 2 - FOG.

## 4.3 Proposed Method

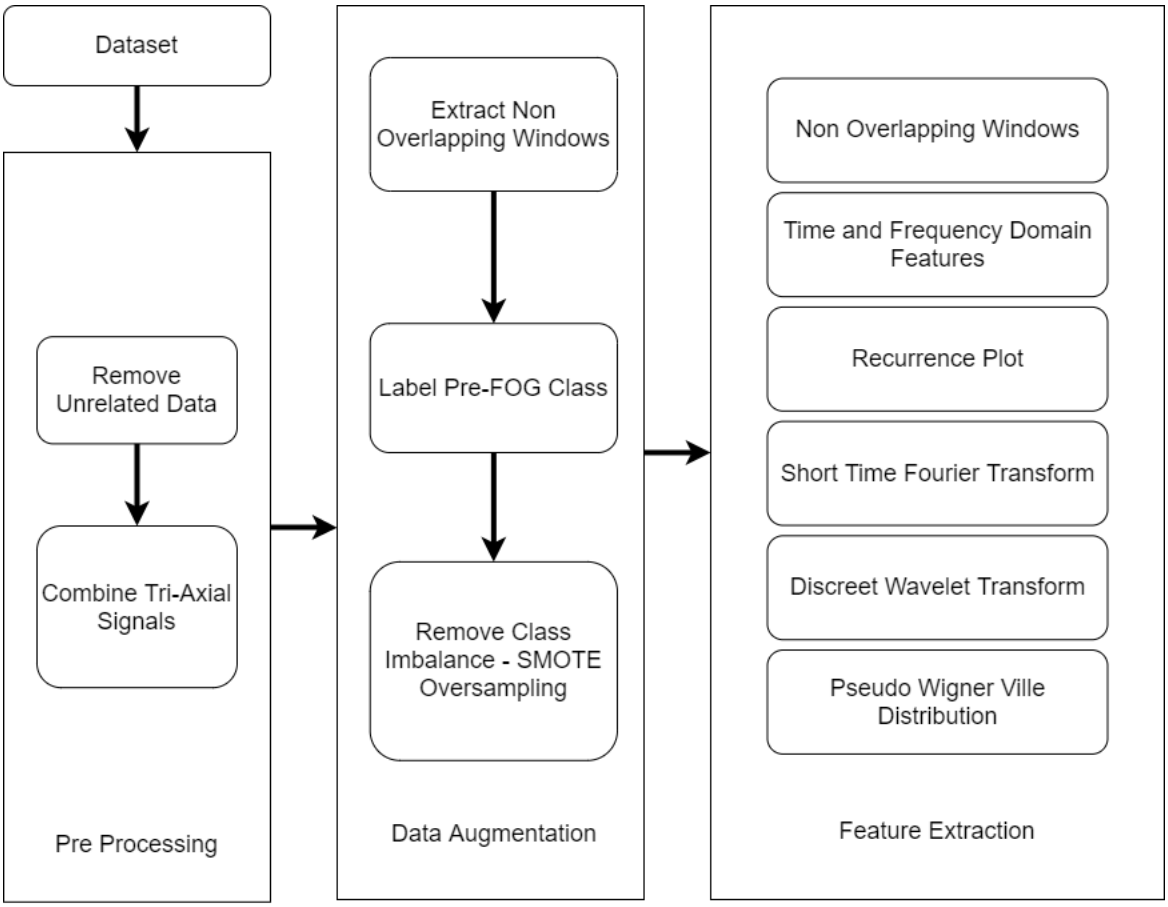



Figure 4.2: Proposed Preprocessing, Data Augmentation and Feature Extraction workflow

The preprocessing, data augmentation and feature extraction algorithm used in this study is presented in Figure 4.2. The major components are explained in the subsections below.

### 4.3.1 Preprocessing

In the preprocessing component, data that is irrelevant to the study is removed and signals from the three axes of each sensor is combined so that there is only

one signal stream for each sensor.

#### 4.3.1.1 Removing Unrelated Data

Data with an annotation of 0 (not a part of the experiment) was removed. We also removed data from subjects who did not experience FOG at all during the experiment.

#### 4.3.1.2 Calculating Magnitude of Acceleration for all three axis

The three signals originating from each channel were combined to calculate the magnitude of acceleration, resulting in three signal streams with one from each of the sensors as shown in Eq. (4.1).

$$\tau_C = \{A_C, L_C, T_C\} \tag{4.1}$$

Magnitude of acceleration is the relative value of the overall acceleration at any given time instance, calculated as shown in Eq. (4.2).

$$\alpha_C = \sqrt{\alpha_X^2 + \alpha_Y^2 + \alpha_Z^2}, where\ \alpha_X, \alpha_Y, \alpha_Z \in \tau,\ \alpha_C \in \tau_C \tag{4.2}$$

Each of the accelerometers was assigned to a single channel, with the data being recorded for three channels: Ankle (A), Leg (L) and Torso (T). Each channel consisted of three separate signals, with each of the signals corresponding to a single axis from the accelerometer. The axes were horizontal forward (X), vertical (Y) and horizontal lateral (Z). Thus, a set $\tau$ of nine signals were recorded for each of the patients with a sampling frequency ($f_s$) of 64, as illustrated in Eq. (4.3).

$$\tau = \{\{A_X, A_Y, A_Z\}, \{L_X, L_Y, L_Z\}, \{T_X, T_Y, T_Z\}\} \tag{4.3}$$

### 4.3.2 Data Augmentation

We applied small non-overlapping windows to extract data from the original continuous signal. The window data immediately before the start of a FOG event was labeled with a new class PreFOG, which is essential for predicting FOG events before they occur. The number of Non-FOG samples vastly

outnumber the PreFOG and FOG samples, making Non-FOG our majority class. In order to solve the issue of class imbalance, the minority classes, PreFOG and FOG, were over sampled to match the number of samples from the Non-FOG class.

#### 4.3.2.1 Signal Segmentation

Non-overlapping 1-dimensional windows of length $f_s \times w$ time-steps were used to extract signal $\alpha_C \in \tau_C$, where $w$ is the length of the signal window in seconds.

$$\alpha_C = [\ \alpha_1\ \alpha_2\ ....\ \alpha_{w \times f_s}\ ]_{w \times f_s} \tag{4.4}$$

Since the windows were non overlapping, shorter window lengths provided a larger dataset. Signals were segmented into window lengths ranging from 1 to 4 seconds. Each signal window generated a separate dataset.

#### 4.3.2.2 Labeling PreFOG class

Mazilu et al. [73] proposed that gait cannot enter into FOG state directly from normal walking without first going through a state of deterioration. They define this state as PreFOG, which is a transition period with variable duration. Identifying this transition state would be valuable for both FOG detection and prediction. Since the duration of PreFOG might not be the same from patient to patient, for our experiment the immediate window ($w \times f_s$ time steps) before the onset of a FOG event was labeled as PreFOG.

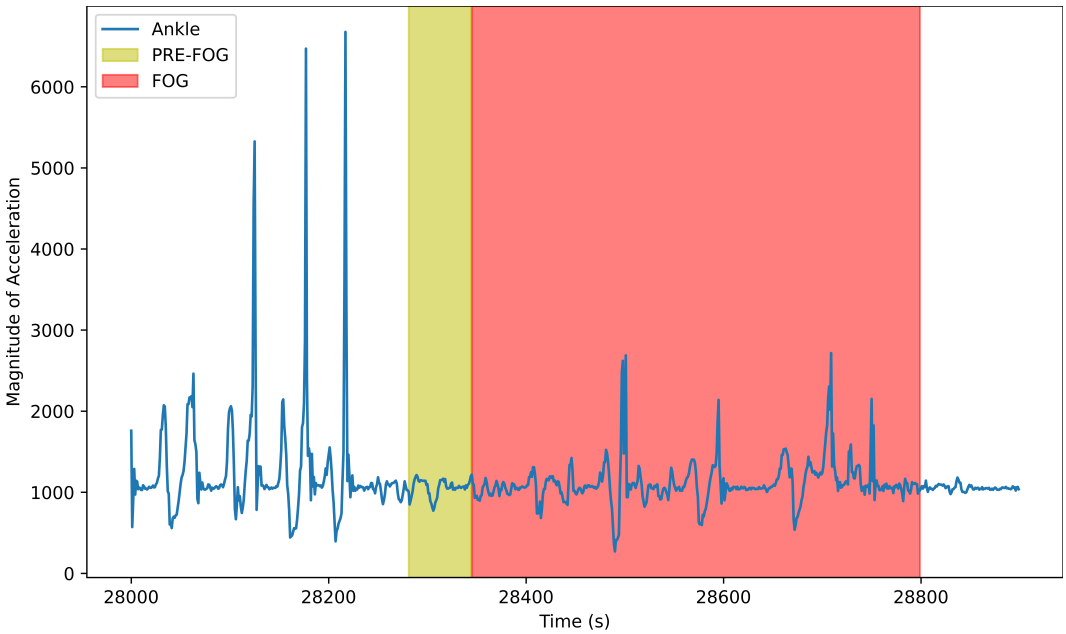



Figure 4.3: Example of combined Accelerometer signal from Ankle, capturing the motor variations in the gait of a Parkinson's patient, containing Normal gait, followed by a window of PreFOG period (Yellow), and then a FOG event (Red).

The final dataset thus had three annotations,

- 0 - Non FOG
- 1 - FOG
- 2 - PreFOG

#### 4.3.2.3 SMOTE Oversampling

At this stage, the dataset was hugely imbalanced, with majority of the data being from the Non-FOG class. Such imbalanced data would lead to most architectures ignoring the minority classes and over-classifying the majority class, although the performance on the minority classes is much more significant in this case. There are multiple ways to address this issue. One approach is to under sample the majority class to match the number of samples in the minority classes. But in our case, the minority samples are sparse, and under sampling the majority class would lead to a drastic decrease in the total

number of training samples. NN architectures require a large number of training samples in order to perform satisfactorily, and therefore under sampling would lead to poor performance. An alternative method is to over sample the minority class. It involves duplicating the samples of the minority class to match the number of samples in the majority class. Although this method balances the class distribution, it does not provide the networks with any new information to learn. We decided to choose the approach proposed by Chawla et al. [16] to synthesize new samples from existing samples. This Synthetic Minority Oversampling Technique (SMOTE) creates new synthetic plausible samples that are in the same feature space as other minority class samples. The generated data was only used for training, the performance of the models was evaluated with real data during testing.

### 4.3.3 Feature Extraction

After data augmentation, our final feature set consisted of 5 different modalities extracted from the same source, $\alpha_i \in \alpha_C$ as shown in Eq. (4.5).

$$Features_i = \{\alpha_i, F_i, RP_i, STFT_i, DWT_i, PWVD_i\}, \alpha_i \in \alpha_C \tag{4.5}$$

where,

- $\alpha_i$ = Moving window extracted from signal $\alpha_C$
- $F_i$ = Manually extracted feature set from $\alpha_i$
- $RP_i$ = Recurrence Plot representation of $\alpha_i$
- $STFT_i$ = Short Time Fourier Transform representation of $\alpha_i$
- $DWT_i$ = Discrete Wavelet Transform representation of $\alpha_i$
- $PWVD_i$ = Pseudo Wigner Ville Distribution representation of $\alpha_i$

#### 4.3.3.1 Time and Frequency Domain Features

For each $\alpha_i \in \alpha_C$, feature relating to the time and frequency domain was extracted, as explained in Table 4.1.

Table 4.1: $F_i$ Features extracted for each $\alpha_i \in \alpha_C$

| **Time Domain Features** | **Description** |
|---|---|
| Min, Max | Minimum and Maximum value of the signal |
| Range | Difference between the minimum and maximum value of the signal |
| Mean | Average value of signal |
| Median | Median value of the signal |
| Mode | Modal value of the signal |
| Trimmed Mean | Trimmed/Truncated mean of the signal |
| Standard Deviation | Deviation of a signal compared to its mean |
| Variance | Square root of the standard deviation of the signal |
| Root mean square | Square root of the mean of the squared signal |
| Mean absolute value | Mean of absolute value of the signal |
| Median absolute deviation | Median over the absolute deviations from the median |
| 25th Percentile | 25th percentile value of the signal |
| 75th Percentile | 75th percentile value of the signal |
| Interquantile range | Difference between the 75th and 25th percentile of the signal |
| Skewness | The degree of asymmetry in the signal |
| Normalized Signal Magnitude Area | Sum of standardized acceleration magnitude normalized by window length |
| Kurtosis | The degree of peakedness in the signal, signals with high kurtosis have more outliers |
| Mean Crossing Rate | The number of times the signals goes from above average value to below average value normalized by the window length |
| Signal Vector Magnitude | Sum of euclidean norm over the window normalized by window length |
| Peak of Fourier Transform | Maximum magnitude of Discrete Fourier Transform of the signal normalized by the window length |
| **Frequency Domain Features** | **Description** |

| | |
|---|---|
| Entropy | Measure of random distribution of frequency |
| Energy | Sum of squared magnitude of FFT of the signal divided by window length |
| Peak Frequency | Maximum frequency value in the power spectrum |
| Freeze Band Power | The sum of power in Freeze band of frequencies divided by sampling frequency |
| Locomotion Band Power | The sum of power in Locomotion band of frequencies divided by sampling frequency |
| Freeze Index | Power of signal in freeze band (3-8Hz) divided by it's Power in locomotion band(0.5-3Hz) |
| Band Power | Sum of the power in freeze band and in locomotion band |

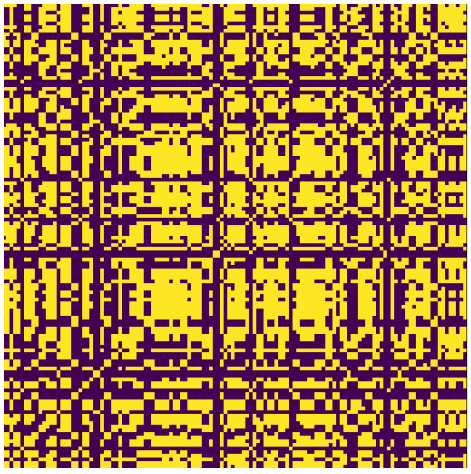
#### 4.3.3.2 Recurrence Plots (RP)

RP are used to represent temporal correlations of univariate series data defined in a square matrix [22]. For time series data, the matrix elements represent the times at which the amplitude of the signal recurs. If $i$ and $j$ are two time instances, and $x(i)$ and $x(j)$ are values in the time series at two recurrence time instances, the formula to compute the recurrence plot [102] is given in Eq. (4.6).

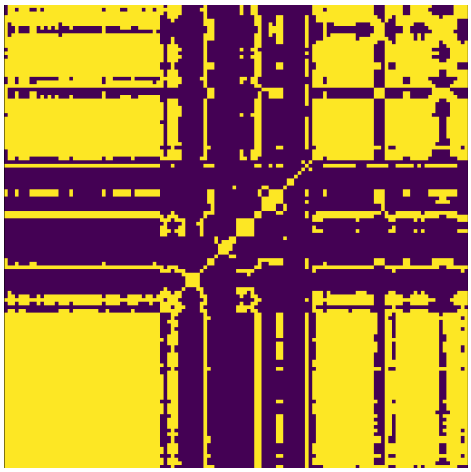
$$R(i,j) = \begin{cases} 1, & \text{if } ||x(i) - x(j) \leq \epsilon|| \\ 0, & \text{otherwise} \end{cases}, (\epsilon \textit{ is a custom similarity threshold}) \tag{4.6}$$

RP are often robust against outliers and noisy data for periodic signals. Some examples of RP for our signals can be seen in Figure 4.4. The plots were generated with a window length ($w$) of 2. It was observed that for $w = 2$, Non-FOG events had no distinct pattern when represented as a recurrence plot, PreFOG events show clear distinct patterns and FOG events had patterns that were more defined than Non-FOG but less defined than PreFOG. Both $x$ and $y$ axes represent time for RP.

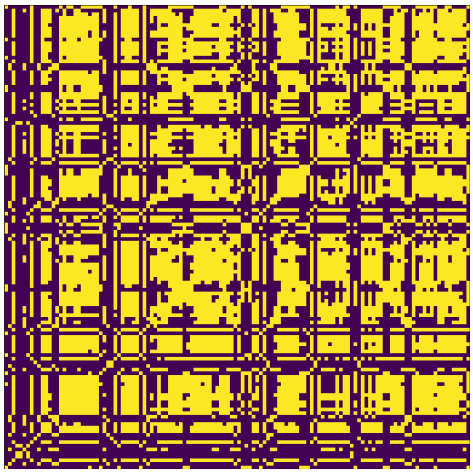
Figure 4.4: Examples of Recurrence Plot generated from Accelerometer signal from Ankle with a Window size of 2 and $\epsilon$s value of 0.5.



(a) Signals representing Normal walking or Non-FOG



(b) Signals representing PreFOG

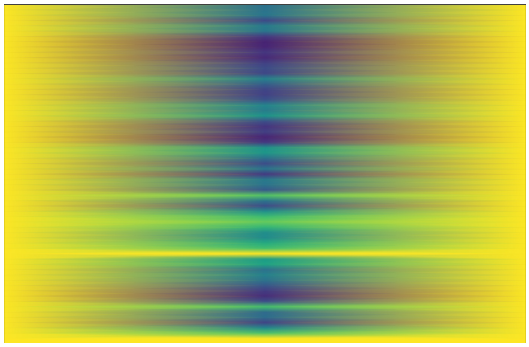


(c) Signals representing FOG

#### 4.3.3.3 Short Time Fourier Transform (STFT)

STFT is a Fourier transform that quantifies the phase content and the sinusoidal frequency of signal segments changing over time [22]. STFT is useful in capturing the time and frequency characteristics in the signals. Rajoub et al. [98] mentioned that STFT does not perform well in capturing sharp signals and patterns with varying duration. Figure 4.5 shows some example spectograms generated using STFT, describing magnitude over time for each of our signal types over a 2 second time window. $x$ axis represents time and $y$ axis frequency for STFT. For $w = 2$, STFT captured the difference between Non-FOG and other classes, with the spectograms for PreFOG and FOG classes being almost clear compared to that of Non-FOG. However, it was difficult to visually differentiate PreFOG and FOG from STFT alone.

Figure 4.5: Examples of Short Time Fourier Transform Plot generated from Accelerometer signal from Ankle with a Window size of 2.



(a) Signals representing Normal walking or Non-FOG

(b) Signals representing PreFOG

(c) Signals representing FOG

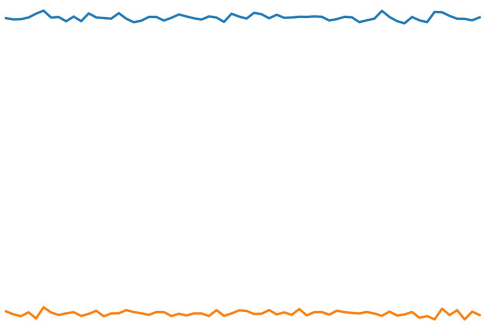
#### 4.3.3.4 Discrete Wavelet Transform (DWT)

DWT is a process of decomposing a signal sequence into subsets, with each subset being a time series consisting coefficients that represent the time evolution in the corresponding frequency band [43]. A main advantage of DWT is the ability to capture both frequency and location characteristics in a time series. Haar Transform is the simplest of wavelet transforms. We used Haar sequence proposed by Haar et al. [35], which is the first known wavelet ba-

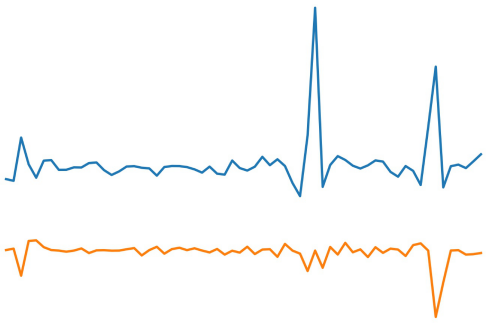
sis. The Haar wavelet can be used to analyze signals with sudden transitions. Figure 4.6 shows sample plots of the approximation and detail coefficients of transforms for a 2 second time window. For DWT, $x$ axis represents time and $y$ axis frequency.

DWT representation plot for $w = 2$ is useful for visually identifying the Non-FOG class compared to PreFOG and FOG classes. For Non-FOG events, the approximation and detail coefficient plots are almost flat, without any large fluctuations in value, which is distinctly identifiable. The representations for PreFOG and FOG events are harder to differentiate as both representations show sudden rise and drop in their values.

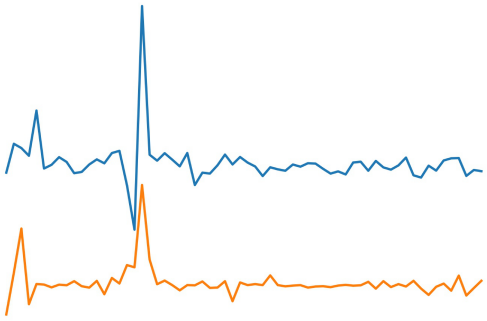
Figure 4.6: Examples of Discrete Wavelet Transformation generated from Accelerometer signal from Ankle with a Window size of 2.



(a) Signals representing Normal walking or Non-FOG



(b) Signals representing PreFOG



(c) Signals representing FOG

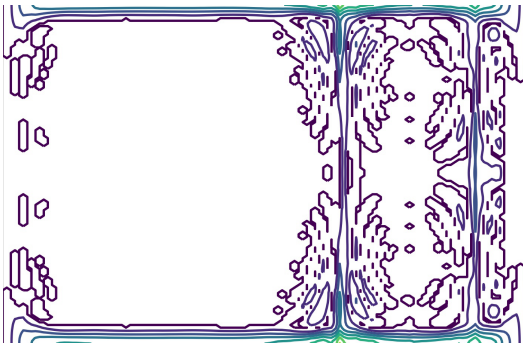
#### 4.3.3.5 Pseudo Wigner Ville Distribution (PWVD)

PWVD is a method to represent transient phenomena in three dimensions, i.e., time, frequency and amplitude [119]. PWVD has been proven to be effective in generating accurate time frequency representation, since its frequency and time resolutions are determined by the resolution of the signals and not by the duration [119]. Figure 4.7 shows some examples of PWVD computed on signals with 2 second time window from our data. For PWVD, $x$ and $y$ axes

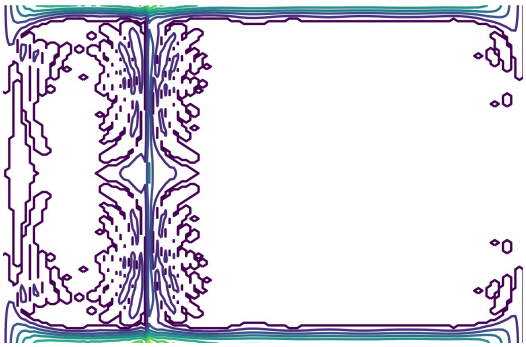
represent time and frequency respectively. The Non-FOG and FOG gaits can be clearly distinguished from PWVD representations for $w = 2$, as Non-FOG gaits have a clear central section compared to FOG events. Both PreFOG and FOG classes have patterns appearing in the central section, which makes it difficult to differentiate them visually.

Figure 4.7: Examples of Pseudo Wigner Ville Distribution generated from Accelerometer signal from Ankle with a Window size of 2.

(a) Signals representing Normal walking or Non-FOG



(b) Signals representing PreFOG



(c) Signals representing FOG

## 4.4 Model Structure

We introduced a CNN based model architecture based on the findings obtained from the four feature visual representations, $RP, STFT, DWT, PWVD$ discussed above. A LSTM based architecture was proposed for the original signal $\alpha$ and the corresponding feature set $F$. For each data modality $\in Features$, an instance of the corresponding model was trained and its performance was recorded. Then, the trained model instances were combined in three ensemble network architectures, $M7$, $M8$ and $M9$, as explained below. Our objective is to demonstrate that ensemble models provide better performance than individual constituent models.

### 4.4.1 Basic Convolutional Neural Network Architecture (CNN)

CNNs are known for their ability to identify complex non-linear relationships between data points without hand crafted feature engineering. To complement our techniques to present time series data visually, a CNN architecture was designed, which is presented in Figure 4.8. The input is passed through four 2D *Convolutional* layers with filter sizes 64, 32, 16 and 8 respectively, a kernel size of $(4, 4)$ and *LeakyReLu* activation function with a negative slope coefficient, and *alpha* value of 0.3. Each of the *Convolutional* layers was followed by a 2D *MaxPooling* layer with a pool size of $(2, 2)$ and a *Dropout* layer having a dropout rate of 0.25. The data was then flattened and passed through two *Dense* layers with 100 and 50 units respectively. Each of the *Dense* layers had *LeakyReLu* activation function with alpha value of 0.3 and was followed by 2 *Dropout* layers having a dropout rate of 0.2. Finally a *Dense* layer with $Softmax$ activation function of 3 units for our three output classes was added. The model was compiled with a $RMSProp$ optimizer with an initial learning rate of 0.0001. The parameter values were selected after trial and error on multiple values. For our four visual feature types, $RP, STFT, DWT, PWVD$, a separate instance model was trained and validated, which are labelled $M1, M2, M3$ and $M4$ respectively.

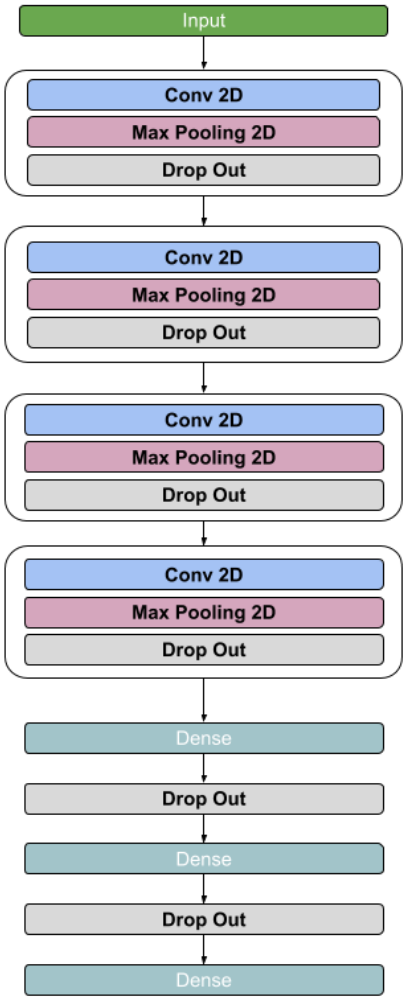



Figure 4.8: Proposed basic CNN Architecture with 4 recurring 2D Convolution blocks, followed by 3 Dense layers.

### 4.4.2 Basic Bidirectional Long Short Term Memory Architecture (LSTM)

LSTM network is a type of recurrent NN architecture, which is suitable for learning and remembering a long sequences of input data, automatically extracting features from the raw sequence and providing comparable performance to using handcrafted features. Bidirectional LSTMs add a duplication of the first recurrent layer. The first layer is trained on the original input sequence and the duplicated layer is trained on a reversed copy of the input sequence. For our data, the use of Bidirectional LSTM is justified because the context

of the whole signal sequence, instead of a linear interpretation, is relevant for FOG identification and prediction. Our Bidirectional LSTM architecture is illustrated in Figure 4.9. The input is passed through four $BidirectionalLSTM$ layers stacked on top of each other with $tanh$ activation function and $n_{layers}$ hidden layers. The value of $n_{layers}$ is computed by Equation (4.7) where $l_{input}$ is the length of the input sequence and $\sigma$ is the multiplication coefficient. The value for $\sigma$ was set to 3 based trial and error as it generated the best results, keeping the rest of the pipeline unchanged. The output of $LSTM$ was passed through a $Dense$ layer with $Softmax$ activation function. The final $Dense$ layer had 3 units to classify between the three output classes. An $Adam$ optimizer with an initial Learning rate of 0.0001 was used to compile the model. One instance of this model, $M5$, was trained on the original signal $\alpha_C$ and another instance, $M6$, was trained on the handcrafted feature set $F_C$ corresponding to the signal $\alpha_C$.

$$n_{layers} = l_{input} \times \sigma \tag{4.7}$$

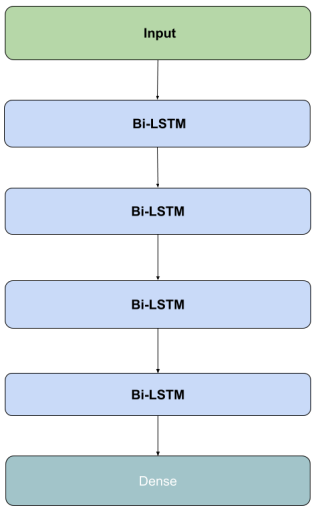



Figure 4.9: Proposed basic Bidirectional LSTM Architecture with 4 recurring Bidirectional LSTM block, followed by a Dense Layer.

### 4.4.3 Ensemble Architectures

Ensemble Learning is a NN training approach, where the predictions from multiple trained networks are combined to solve a problem [143]. In this

work, three ensemble network architectures are examined (Figure 4.10). The constituent model set is defined as,

$$M_{constituent} = \{M1, M2, M3, M4, M5, M6\} \tag{4.8}$$

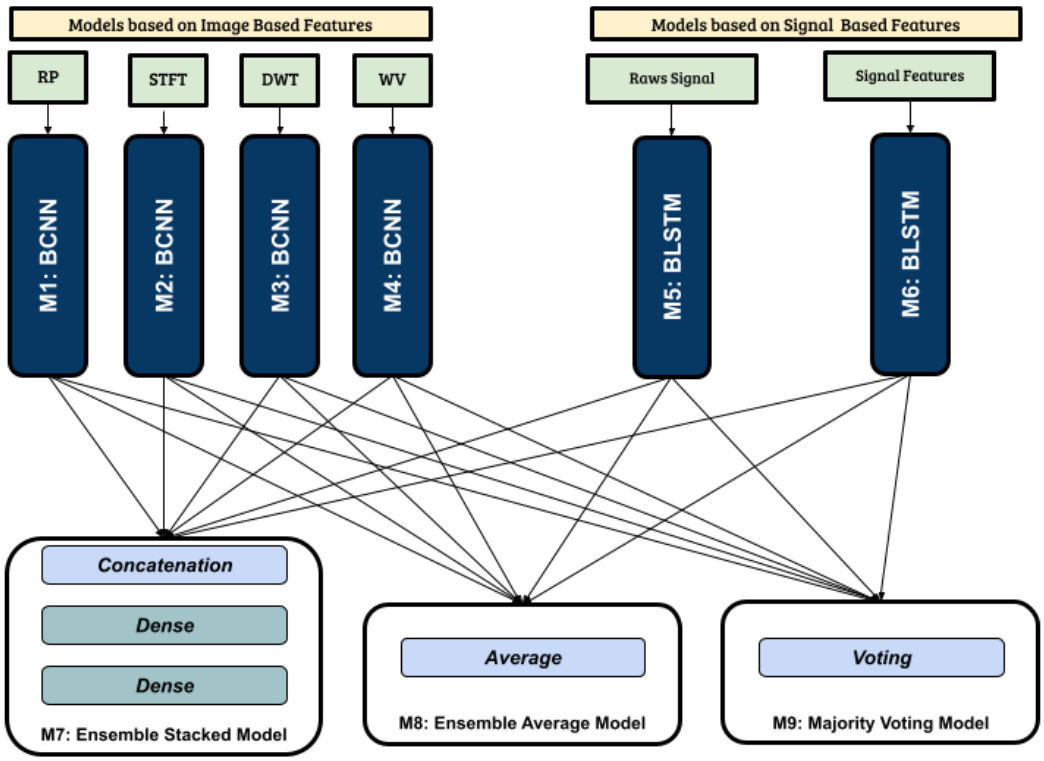



Figure 4.10: Proposed Ensemble Architectures, (1) M7 concatenates the output all constituent models, followed by a Dense Layer, (2) M8 Averages the outputs of all constituent models and (3) M9 calculates the majority prediction of all models using mode.

#### 4.4.3.1 Stacked Ensemble Model - M7:

This model architecture is designed by combining the output predictions of all $M_i \in M_{constituent}$. The models have already been trained on their respective data, and all layers of constituent models are set as non-trainable before adding them to the ensemble model. The constituent models were already proven to produce good results and our purpose was to compare the performance of ensemble approaches keeping the constituent models unchanged. Thus they were set to be non-trainable. The outputs of the models are passed through a

*Concatenation* layer and then two *Dense* layers with 10 and 3 units respectively. The first *Dense* layer has a *ReLu* activation function and the final *Dense* layer has a $Softmax$ activation function. An *Adam* optimizer is used with a learning rate of 0.0001.

#### 4.4.3.2 Average Ensemble Model - M8:

This model architecture takes the average of the predicted outputs of all $M_i \in M_{constituent}$. The constituent models pre-trained on their respective data are set as non trainable, and the outputs are passed through an *Average* layer. $M8$ is compiled with an *Adam* optimizer having a learning rate of 0.0001.

#### 4.4.3.3 Majority Voting - M9:

For majority voting, the output is predicted as the majority class predicted by the constituent models $M_i \in M_{constituent}$. The hard voting approach is used to calculate the final outcome, where every constituent model votes for an output class and the majority vote is selected as the final prediction. In statistical terms, this is equivalent to calculating the Mode of the predictions from all constituent models.

## 4.5 Results

### 4.5.1 K-Fold Cross Validation

In order to get an accurate estimate of the models performance, Stratified K-fold cross validation technique was utilized to separate the data into training and testing sets, with 80% of the data being used for training and the rest for testing, preserving the ratio of samples of each class. Since neural network models take a long time to train and evaluate, it is difficult to use high values for K. For this experiment, K was set to 5. The dataset was first shuffled, and then it was split into K unique class balanced (train, test) combinations. For each fold, a new instance of each of our models was trained using the training set and its performance on the testing set was evaluated and recorded. The evaluation performances were retained while the instances of the models were

discarded. Finally, the average performance of the models across all K folds was recorded.

### 4.5.2 Normalization

The visual feature representations in $RP_i, STFT_i, DWT_i, PWVD_i$ are normalized using Eq. (4.9). Since the range for RGB values in images is $0 - 255$, each channel is normalized to the range of $0.0 - 1.0$. Then the values are centered through division with the mean.

$$p_{normalized} = \frac{p_o - 255.0}{mean(p_o)}, p_o \in RP_i, STFT_i, DWT_i, PWVD_i \tag{4.9}$$

### 4.5.3 Experimental Setup

For each of the training sets, it was further divided into (train,validation) sets with 80% being used for training and the rest for validation. Validation using unseen data was crucial to evaluate whether the model was learning over time by comparing its performances. Each of the models was trained for at most 500 epochs. The training was stopped if the validation accuracy did not improve over 50 epochs. All the experiments were done on a Ubuntu Machine, with 4 core Intel Xeon Processor, 62 Gigabytes of RAM and Nvidia Tesla GPU with 16 Gigabytes memory. The algorithm was tested with data from all three wearable sensors $\alpha_C \in \tau_C$, but we achieved the best performance using only data from the ankle mounted sensor $A_C$. All results presented here are based on $A_C$. In order to train Convolutional Neural Networks, all image features were adjusted to the shape of (3, 128, 128) and then normalized. The runtimes presented include both training and testing of the model but do not include the preprocessing and feature extraction time. Runtimes presented here for ensemble models do not include the time for training the constituent models. All scores presented in Section 4.5.4 are average scores. Five instances of each model were trained and evaluated on five folds of (train, test) sets, and their scores and runtimes were averaged.

### 4.5.4 Metric Scores

Table 4.2: Results of some existing methods

| **Model** | **Window length (s)** | **Recall/ Sensitivity** | **Specificity** | $F_\beta$ **Score** |
|---|---|---|---|---|
| Mazilu et al.[73] (Unsupervised - 20 Features) | 3 | 76.86 | 86.21 | 81.56 |
| Mazilu et al.[73] (Supervised - 20 Features) | 3 | 66.65 | 88.74 | 78.27 |
| Mazilu et al.[73] (Unsupervised - 25 Features) | 3 | 76.86 | 85.52 | 80.82 |
| Mazilu et al.[73] (Supervised - 25 Features) | 3 | 67.58 | 88.52 | 78.65 |
| Decision Tree [33] | 4 | 96.70 | 98.92 | - |
| Random Forest [33] | 4 | 98.91 | 99.44 | - |
| AdaBoost [33] | 4 | 97.99 | 99.56 | - |
| KNN [33] | 4 | 94.61 | 97.38 | - |
| SVM [33] | 4 | 97.54 | 98.64 | - |
| ProtoNN [33] | 4 | 95.25 | 99.66 | - |
| Bonsai [33] | 4 | 92.9 | 98.36 | - |

Table 4.3: Results of Basic CNN architecture M1 with RP

| **Data Type** | **Window (s)** | **Accuracy** | **Precision** | **Recall/ Sensitivity** | **Specificity** | $F_\beta$ **Score** | **MCC** | **Runtime (Min)** |
|---|---|---|---|---|---|---|---|---|
| $A_C$ (4.1) | 2 | **0.894 ± 0.021** | **0.929 ± 0.006** | **0.894 ± 0.021** | **0.933 ± 0.007** | **0.904 ± 0.016** | 0.687 ± 0.039 | 52.05 |
| | 3 | 0.832 ± 0.023 | 0.905 ± 0.011 | 0.832 ± 0.023 | 0.926 ± 0.010 | 0.849 ± 0.021 | 0.648 ± 0.035 | 17:41 |
| | 4 | 0.864 ± 0.030 | 0.901 ± 0.020 | 0.864 ± 0.030 | 0.925 ± 0.020 | 0.873 ± 0.027 | **0.695 ± 0.058** | **11:52** |
| $L_C$ (4.1) | 2 | **0.891 ± 0.019** | **0.929 ± 0.008** | **0.891 ± 0.019** | **0.937 ± 0.008** | **0.902 ± 0.01**6 | 0.684 ± 0.037 | 57:31 |
| | 3 | 0.873 ± 0.005 | 0.915 ± 0.006 | 0.873 ± 0.005 | 0.937 ± 0.006 | 0.883 ± 0.005 | **0.702 ± 0.014** | 22:41 |
| | 4 | 0.840 ± 0.022 | 0.895 ± 0.013 | 0.840 ± 0.022 | 0.930 ± 0.014 | 0.853 ± 0.020 | 0.664 ± 0.037 | **9:25** |
| $T_C$ (4.1) | 2 | **0.926 ± 0.013** | **0.942 ± 0.007** | **0.926 ± 0.013** | **0.944 ± 0.009** | **0.930 ± 0.011** | **0.755 ± 0.033** | 64:58 |
| | 3 | 0.876 ± 0.021 | 0.921 ± 0.008 | 0.876 ± 0.021 | 0.924 ± 0.008 | 0.889 ± 0.017 | 0.649 ± 0.038 | 15:24 |
| | 4 | 0.849 ± 0.000 | 0.911 ± 0.000 | 0.849 ± 0.000 | 0.934 ± 0.000 | 0.863 ± 0.000 | 0.667 ± 0.000 | **9:59** |

Table 4.4: Results of Basic CNN architecture M2 with STFT

| **Data Type** | **Window (s)** | **Accuracy** | **Precision** | **Recall/ Sensitivity** | **Specificity** | $F_\beta$ **Score** | **MCC** | **Runtime (Min)** |
|---|---|---|---|---|---|---|---|---|
| $A_C$ (4.1) | 2 | 0.678 ± 0.016 | **0.893 ± 0.008** | 0.678 ± 0.0176 | 0.860 ± 0.007 | 0.742 ± 0.011 | 0.407 ± 0.014 | 68.11 |
| | 3 | **0.799 ± 0.030** | 0.883 ± 0.010 | **0.799 ± 0.030** | 0.897 ± 0.012 | **0.819 ± 0.026** | 0.585 ± 0.038 | 18:41 |
| | 4 | 0.781 ± 0.030 | 0.877 ± 0.005 | 0.781 ± 0.030 | **0.905 ± 0.01**0 | 0.803 ± 0.025 | **0.588 ± 0.026** | **11:22** |
| $L_C$ (4.1) | 2 | 0.719 ± 0.023 | 0.889 ± 0.009 | 0.719 ± 0.023 | 0.864 ± 0.017 | 0.762 ± 0.019 | 0.452 ± 0.035 | 127:59 |
| | 3 | **0.815 ± 0.025** | **0.902 ± 0.006** | **0.815 ± 0.025** | **0.921 ± 0.008** | **0.834 ± 0.021** | **0.633 ± 0.030** | 21:02 |
| | 4 | 0.746 ± 0.036 | 0.865 ± 0.010 | 0.746 ± 0.036 | 0.894 ± 0.013 | 0.770 ± 0.031 | 0.550 ± 0.032 | **8:55** |
| $T_C$ (4.1) | 2 | 0.781 ± 0.010 | 0.899 ± 0.002 | 0.781 ± 0.010 | 0.894 ± 0.003 | 0.813 ± 0.008 | 0.516 ± 0.008 | 57:19 |
| | 3 | **0.831 ± 0.037** | **0.911 ± 0.013** | **0.831 ± 0.037** | 0.914 ± 0.019 | **0.854 ± 0.030** | 0.586 ± 0.060 | 14:36 |
| | 4 | 0.816 ± 0.005 | 0.905 ± 0.004 | 0.816 ± 0.005 | **0.927 ± 0.003** | 0.836 ± 0.004 | **0.629 ± 0.004** | **8:19** |

Table 4.5: Results of Basic CNN architecture M3 with DWT

| **Data Type** | **Window (s)** | **Accuracy** | **Precision** | **Recall/ Sensitivity** | **Specificity** | $F_\beta$ **Score** | **MCC** | **Runtime (Min)** |
|---|---|---|---|---|---|---|---|---|
| $A_C$ (4.1) | 2 | **0.939 ± 0.002** | **0.948 ± 0.004** | **0.939 ± 0.002** | 0.947 ± 0.009 | **0.942 ± 0.002** | 0.785 ± 0.013 | 67.44 |
| | 3 | 0.922 ± 0.031 | 0.940 ± 0.019 | 0.922 ± 0.031 | **0.949 ± 0.017** | 0.926 ± 0.028 | **0.796 ± 0.068** | 22:39 |
| | 4 | 0.906 ± 0.022 | 0.923 ± 0.018 | 0.906 ± 0.022 | 0.941 ± 0.017 | 0.910 ± 0.021 | 0.766 ± 0.050 | **13:48** |
| $L_C$ (4.1) | 2 | 0.923 ± 0.005 | 0.941 ± 0.001 | 0.923 ± 0.005 | 0.944 ± 0.006 | 0.928 ± 0.004 | 0.748 ± 0.003 | 127:32 |
| | 3 | **0.940 ± 0.010** | **0.951 ± 0.006** | **0.940 ± 0.010** | **0.963 ± 0.003** | **0.943 ± 0.009** | **0.834 ± 0.020** | 26:43 |
| | 4 | 0.930 ± 0.020 | 0.940 ± 0.015 | 0.930 ± 0.020 | 0.958 ± 0.014 | 0.932 ± 0.019 | 0.817 ± 0.044 | **16:47** |
| $T_C$ (4.1) | 2 | **0.946 ± 0.015** | **0.954 ± 0.010** | **0.946 ± 0.015** | 0.952 ± 0.010 | **0.949 ± 0.013** | 0.811 ± 0.042 | 55:17 |
| | 3 | 0.938 ± 0.019 | 0.949 ± 0.012 | 0.938 ± 0.019 | **0.952 ± 0.008** | 0.941 ± 0.017 | 0.784 ± 0.048 | 22:12 |
| | 4 | 0.943 ± 0.008 | 0.953 ± 0.005 | 0.943 ± 0.008 | 0.971 ± 0.003 | 0.945 ± 0.008 | **0.839 ± 0.019** | **12:39** |

Table 4.6: Results of Basic CNN architecture M4 with PWVD

| **Data Type** | **Window (s)** | **Accuracy** | **Precision** | **Recall/ Sensitivity** | **Specificity** | $F_\beta$ **Score** | **MCC** | **Runtime (Min)** |
|---|---|---|---|---|---|---|---|---|
| $A_C$ (4.1) | 2 | 0.831 ± 0.023 | 0.906 ± 0.006 | 0.831 ± 0.023 | 0.902 ± 0.011 | 0.852 ± 0.018 | 0.571 ± 0.035 | 74.00 |
| | 3 | **0.865 ± 0.015** | **0.907 ± 0.009** | **0.865 ± 0.015** | **0.930 ± 0.010** | **0.876 ± 0.014** | **0.681 ± 0.031** | 32:34 |
| | 4 | 0.825 ± 0.024 | 0.888 ± 0.010 | 0.825 ± 0.024 | 0.919 ± 0.010 | 0.839 ± 0.021 | 0.641 ± 0.035 | **17:46** |
| $L_C$ (4.1) | 2 | 0.811 ± 0.011 | 0.901 ± 0.005 | 0.811 ± 0.011 | 0.897 ± 0.009 | 0.836 ± 0.009 | 0.545 ± 0.021 | 131:00 |
| | 3 | **0.871 ± 0.020** | **0.912 ± 0.005** | **0.871 ± 0.020** | **0.930 ± 0.005** | **0.881 ± 0.016** | **0.695 ± 0.026** | 28:58 |
| | 4 | 0.831 ± 0.025 | 0.895 ± 0.010 | 0.831 ± 0.025 | 0.923 ± 0.013 | 0.846 ± 0.021 | 0.657 ± 0.032 | **16:27** |
| $T_C$ (4.1) | 2 | 0.805 ± 0.011 | 0.900 ± 0.010 | 0.805 ± 0.011 | 0.887 ± 0.017 | 0.832 ± 0.010 | 0.531 ± 0.036 | 68:28 |
| | 3 | 0.842 ± 0.033 | 0.908 ± 0.009 | 0.842 ± 0.033 | 0.917 ± 0.013 | 0.861 ± 0.026 | 0.590 ± 0.052 | 29:24 |
| | 4 | **0.864 ± 0.025** | **0.916 ± 0.010** | **0.864 ± 0.025** | **0.946 ± 0.012** | **0.877 ± 0.021** | **0.687 ± 0.040** | **15:50** |

Table 4.7: Results of Bidirectional LSTM architecture M5 with Raw Signals

| **Data Type** | **Window (s)** | **Accuracy** | **Precision** | **Recall/ Sensitivity** | **Specificity** | $F_\beta$ **Score** | **MCC** | **Runtime (Min)** |
|---|---|---|---|---|---|---|---|---|
| $A_C$ (4.1) | 2 | **0.797 ± 0.106** | 0.896 ± 0.027 | **0.797 ± 0.106** | 0.917 ± 0.048 | **0.827 ± 0.082** | 0.527 ± 0.149 | 302.21 |
| | 3 | 0.784 ± 0.124 | **0.903 ± 0.018** | 0.784 ± 0.124 | **0.939 ± 0.053** | 0.817 ± 0.092 | **0.597 ± 0.152** | 149:31 |
| | 4 | 0.695 ± 0.178 | 0.832 ± 0.053 | 0.695 ± 0.178 | 0.774 ± 0.074 | 0.715 ± 0.155 | 0.441 ± 0.168 | **73:22** |
| $L_C$ (4.1) | 2 | 0.705 ± 0.106 | 0.877 ± 0.012 | 0.705 ± 0.106 | 0.836 ± 0.034 | 0.747 ± 0.088 | 0.418 ± 0.091 | 121:56 |
| | 3 | **0.894 ± 0.021** | **0.929 ± 0.006** | **0.894 ± 0.021** | **0.933 ± 0.007** | **0.904 ± 0.016** | **0.687 ± 0.039** | 96.09 |
| | 4 | 0.360 ± 0.380 | 0.315 ± 0.432 | 0.360 ± 0.380 | 0.806 ± 0.134 | 0.312 ± 0.419 | 0.246 ± 0.351 | 52:05 |
| $T_C$ (4.1) | 2 | 0.715 ± 0.270 | 0.880 ± 0.071 | 0.715 ± 0.270 | 0.800 ± 0.120 | 0.744 ± 0.239 | 0.478 ± 0.314 | 199:10 |
| | 3 | **0.778 ± 0.142** | **0.903 ± 0.032** | **0.778 ± 0.142** | **0.930 ± 0.060** | **0.814 ± 0.111** | **0.531 ± 0.181** | 105:22 |
| | 4 | 0.441 ± 0.312 | 0.517 ± 0.372 | 0.441 ± 0.312 | 0.728 ± 0.059 | 0.411 ± 0.305 | 0.111 ± 0.157 | **74:20** |

Table 4.8: Results of Bidirectional LSTM architecture M6 with Extracted features

| **Data Type** | **Window (s)** | **Accuracy** | **Precision** | **Recall/ Sensitivity** | **Specificity** | $F_\beta$ **Score** | **MCC** | **Runtime (Min)** |
|---|---|---|---|---|---|---|---|---|
| $A_C$ (4.1) | 2 | **0.846 ± 0.034** | **0.912 ± 0.015** | **0.846 ± 0.034** | 0.902 ± 0.023 | **0.865 ± 0.028** | 0.600 ± 0.067 | 179.36 |
| | 3 | 0.815 ± 0.020 | 0.896 ± 0.007 | 0.815 ± 0.020 | **0.923 ± 0.009** | 0.834 ± 0.017 | **0.620 ± 0.026** | 91:14 |
| | 4 | 0.803 ± 0.033 | 0.875 ± 0.010 | 0.803 ± 0.033 | 0.893 ± 0.009 | 0.821 ± 0.028 | 0.597 ± 0.040 | **49:37** |
| $L_C$ (4.1) | 2 | **0.822 ± 0.013** | **0.907 ± 0.008** | **0.822 ± 0.013** | 0.904 ± 0.012 | **0.845 ± 0.010** | 0.563 ± 0.025 | 147:41 |
| | 3 | 0.812 ± 0.011 | 0.897 ± 0.001 | 0.812 ± 0.011 | **0.917 ± 0.002** | 0.832 ± 0.008 | **0.620 ± 0.011** | 89:48 |
| | 4 | 0.783 ± 0.070 | 0.859 ± 0.042 | 0.783 ± 0.070 | 0.855 ± 0.043 | 0.801 ± 0.062 | 0.557 ± 0.122 | **47:42** |
| $T_C$ (4.1) | 2 | **0.840 ± 0.007** | **0.907 ± 0.014** | **0.840 ± 0.007** | **0.898 ± 0.025** | **0.859 ± 0.006** | **0.579 ± 0.039** | 190:06 |
| | 3 | 0.773 ± 0.022 | 0.900 ± 0.001 | 0.773 ± 0.022 | 0.897 ± 0.004 | 0.807 ± 0.017 | 0.519 ± 0.019 | 61:44 |
| | 4 | 0.770 ± 0.019 | 0.888 ± 0.013 | 0.770 ± 0.019 | 0.909 ± 0.019 | 0.797 ± 0.017 | 0.560 ± 0.035 | **40:11** |

Table 4.9: Results of Stacked Ensemble M7 with Extracted features

| **Data Type** | **Window (s)** | **Accuracy** | **Precision** | **Recall/ Sensitivity** | **Specificity** | $F_\beta$ **Score** | **MCC** | **Runtime (Min)** |
|---|---|---|---|---|---|---|---|---|
| $A_C$ (4.1) | 2 | 0.971 ± 0.007 | 0.972 ± 0.007 | 0.971 ± 0.007 | 0.956 ± 0.012 | 0.971 ± 0.007 | 0.885 ± 0.027 | 200.32 |
| | 3 | **0.979 ± 0.002** | **0.979 ± 0.002** | **0.979 ± 0.002** | **0.977 ± 0.005** | **0.979 ± 0.002** | **0.934 ± 0.006** | 157:30 |
| | 4 | 0.967 ± 0.005 | 0.967 ± 0.007 | 0.967 ± 0.005 | 0.967 ± 0.012 | 0.967 ± 0.006 | 0.905 ± 0.018 | **108:52** |
| $L_C$ (4.1) | 2 | 0.967 ± 0.008 | 0.968 ± 0.008 | 0.967 ± 0.008 | 0.954 ± 0.013 | 0.967 ± 0.008 | 0.870 ± 0.032 | 200:29 |
| | 3 | **0.980 ± 0.002** | **0.980 ± 0.002** | **0.980 ± 0.002** | **0.977 ± 0.005** | **0.980 ± 0.002** | **0.938 ± 0.006** | 132:45 |
| | 4 | 0.965 ± 0.011 | 0.965 ± 0.011 | 0.965 ± 0.011 | 0.968 ± 0.014 | 0.965 ± 0.011 | 0.899 ± 0.032 | **118:05** |
| $T_C$ (4.1) | 2 | **0.972 ± 0.009** | **0.973 ± 0.009** | **0.972 ± 0.009** | 0.956 ± 0.013 | **0.972 ± 0.009** | 0.889 ± 0.036 | 324:54 |
| | 3 | 0.971 ± 0.006 | 0.971 ± 0.006 | 0.971 ± 0.006 | **0.960 ± 0.008** | 0.971 ± 0.006 | 0.882 ± 0.024 | 196:53 |
| | 4 | 0.967 ± 0.009 | 0.971 ± 0.007 | 0.967 ± 0.009 | 0.979 ± 0.003 | 0.968 ± 0.009 | **0.900 ± 0.026** | **178:51** |

Table 4.10: Results of Average Ensemble M8 with Extracted features

| **Data Type** | **Window (s)** | **Accuracy** | **Precision** | **Recall/ Sensitivity** | **Specificity** | $F_\beta$ **Score** | **MCC** | **Runtime (Min)** |
|---|---|---|---|---|---|---|---|---|
| $A_C$ (4.1) | 2 | 0.979 ± 0.008 | 0.978 ± 0.009 | 0.979 ± 0.008 | 0.958 ± 0.013 | 0.978 ± 0.009 | 0.913 ± 0.034 | 103.09 |
| | 3 | **0.980 ± 0.005** | **0.980 ± 0.005** | **0.980 ± 0.005** | **0.976 ± 0.007** | **0.980 ± 0.005** | **0.938 ± 0.015** | 43:36 |
| | 4 | 0.967 ± 0.005 | 0.967 ± 0.006 | 0.967 ± 0.005 | 0.967 ± 0.012 | 0.967 ± 0.006 | 0.905 ± 0.018 | **32:09** |
| $L_C$ (4.1) | 2 | 0.973 ± 0.008 | 0.973 ± 0.008 | 0.973 ± 0.008 | 0.956 ± 0.013 | 0.973 ± 0.008 | 0.893 ± 0.032 | 107:14 |
| | 3 | **0.978 ± 0.003** | **0.978 ± 0.002** | **0.978 ± 0.003** | **0.976 ± 0.003** | **0.978 ± 0.003** | **0.931 ± 0.008** | 51:31 |
| | 4 | 0.969 ± 0.008 | 0.970 ± 0.008 | 0.969 ± 0.008 | 0.969 ± 0.013 | 0.969 ± 0.008 | 0.911 ± 0.024 | **24:04** |
| $T_C$ (4.1) | 2 | **0.983 ± 0.006** | **0.983 ± 0.006** | **0.983 ± 0.006** | **0.960 ± 0.012** | **0.983 ± 0.006** | **0.932 ± 0.026** | 64:56 |
| | 3 | 0.975 ± 0.005 | 0.975 ± 0.005 | 0.975 ± 0.005 | 0.962 ± 0.009 | 0.975 ± 0.005 | 0.900 ± 0.019 | 32:55 |
| | 4 | 0.976 ± 0.008 | 0.978 ± 0.007 | 0.976 ± 0.008 | 0.983 ± 0.003 | 0.976 ± 0.008 | 0.925 ± 0.024 | **38:40** |

Table 4.11: Results of Majority Voting M9 with Extracted features

| **Data Type** | **Window (s)** | **Accuracy** | **Precision** | **Recall/ Sensitivity** | **Specificity** | $F_\beta$ **Score** | **MCC** | **Runtime (Min)** |
|---|---|---|---|---|---|---|---|---|
| $A_C$ (4.1) | 2 | 0.981 ± 0.007 | 0.980 ± 0.007 | 0.981 ± 0.007 | 0.951 ± 0.015 | 0.980 ± 0.007 | 0.921 ± 0.029 | $< 1$ |
| | 3 | **0.985 ± 0.003** | **0.985 ± 0.003** | **0.985 ± 0.003** | **0.979 ± 0.006** | **0.985 ± 0.003** | **0.953 ± 0.010** | $< 1$ |
| | 4 | 0.969 ± 0.006 | 0.969 ± 0.007 | 0.969 ± 0.006 | 0.967 ± 0.012 | 0.969 ± 0.007 | 0.911 ± 0.019 | $< 1$ |
| $L_C$ (4.1) | 2 | **0.977 ± 0.008** | **0.977 ± 0.008** | **0.977 ± 0.008** | 0.958 ± 0.012 | **0.977 ± 0.008** | 0.907 ± 0.032 | $< 1$ |
| | 3 | 0.973 ± 0.008 | 0.975 ± 0.006 | 0.973 ± 0.008 | **0.974 ± 0.002** | 0.973 ± 0.007 | **0.917 ± 0.020** | $< 1$ |
| | 4 | 0.971 ± 0.008 | 0.972 ± 0.008 | 0.971 ± 0.008 | 0.967 ± 0.008 | 0.971 ± 0.008 | 0.917 ± 0.023 | $< 1$ |
| $T_C$ (4.1) | 2 | **0.983 ± 0.007** | **0.983 ± 0.007** | **0.983 ± 0.007** | 0.960 ± 0.012 | **0.983 ± 0.007** | **0.932 ± 0.030** | $< 1$ |
| | 3 | 0.977 ± 0.003 | 0.977 ± 0.004 | 0.977 ± 0.003 | 0.962 ± 0.008 | 0.976 ± 0.004 | 0.905 ± 0.015 | $< 1$ |
| | 4 | 0.976 ± 0.011 | 0.978 ± 0.009 | 0.976 ± 0.011 | **0.979 ± 0.004** | 0.976 ± 0.010 | 0.925 ± 0.032 | $< 1$ |

## 4.6 Discussion

Since we used non-overlapping time windows, smaller window sizes yielded significantly larger amount of data, which led to better performance in neural network based architectures. We experimented with widow sizes of 2, 3, 4 seconds. A window size of $w$ seconds means that our model is able to predict the start of a FOG event $w$ seconds before it happens. We believe that a window size of 1 would lead to much better detection performance since it means more training examples for the model. But we did not use a smaller window size of 1 because it would also decrease the time window by which we can predict the FOG event. Larger amount of data would also lead to a higher resource consumption during training.

We also observed that the size to which the features in $RP_i$, $STFT_i$, $DWT_i$, $PWVD_i$ are reshaped also plays a vital role in model performance, with larger sizes producing better results. Due to resource constraints, we set this size to be $(3, 128, 128)$. We considered smaller dimensions as well, but they did not yield better results.

Section 4.5.4 presents the performance of for each model $M \in M_1, .....M_9$ with signal $S \in \tau_C$ (for each of $Ankle(A_C)$, $Leg(L_C)$ and $Trunk(T_C)$) with Window Size $w \in 2, 3, 4(seconds)$. The tables report multiple evaluation metric scores including Accuracy, Precision, Recall/Sensitivity, Specificity, $F_\beta$ score, MCC score and the Runtime taken for the model to train in minutes. All scores are reported in the range of (0, 1), except MCC score, which is in the range of (-1, 1). The scores are reported in $Mean \pm Standard\,Deviation$ format. The best scores for each model using the same modality of data but with different window sizes were reported in bold font.

From the metric scores presented in Table 4.3, it can be seen that Basic CNN M1 trained on RP generated from signals performs reasonably well across all metric scores. In most cases the smallest window size of 2 seconds yielded the best scores, but there was no drastic decrease in performance when we increased the window sizes. Comparing sensor locations, data collected from Trunk sensor ($T_C$) performed the best, followed closely by data collected from

Ankle ($A_C$) and Leg ($L_C$).

Table 4.4 presents the scores for Basic CNN M2 trained on STFT plots generated from the signals. For STFT. window size of 3 seemed to provide comparatively better results, although the scores were poor when compared to the scores from RP. The data collected from Trunk sensor ($T_C$) provided best results when using STFT, followed closely by data collected from Leg ($L_C$) and Ankle ($A_C$).

The metric scores of Basic CNN M3 using DWT are reported in Table 4.5. M3 achieved the highest accuracy among our models using visual features (RP, STFT, DWT, PWVD). The scores for varying window sizes were very similar, with a window size of 2 seconds providing the best scores for Ankle ($A_C$) and Trunk ($T_C$) sensor data. For data collected from the Leg ($L_C$), a window size of 3 generated the best scores. Comparing the scores of the three sensors locations, it was noted that Trunk ($T_C$) provided the best scores, followed very close by Leg ($L_C$) and Ankle ($A_C$).

Table 4.6 notes the metric scores of Basic CNN M4 using PWVD. A window size of 3 seconds provided the best scores for Ankle ($A_C$) and Leg ($L_C$) sensor data. For data collected from the Trunk ($T_C$), a window size of 4 generated the best scores. Comparing the best scores for each sensor location, the scores for all three locations were pretty similar with there being no clear advantages.

Table 4.7 contains the scores of Bidirectional LSTM with extracted raw signal windows. The overall performance is not as as good as using visual features. The performance do not experience drastic changes when the window size increases from 2 to 3 seconds, but we see significant drop in performance as the window size changes from 3 to 4 seconds. A window size of 3 seconds provided comparatively better scores for Trunk ($T_C$) and Leg ($L_C$) sensor data. For data collected from the Ankle ($A_C$), a window size of 2 generated the best scores. Data from Leg ($L_C$) sensor provided the best overall scores when using bidirectional LSTM and raw signals.

The scores of Bidirectional LSTM with extracted features are presented in Table 4.8. The performance is slightly better than using raw signals. The performance does not experience drastic changes with changes in the window

size. A window size of 2 seconds provided comparatively better scores for Ankle ($A_C$), Trunk ($T_C$) and Leg ($L_C$) sensor data. Data from Ankle ($A_C$) sensor provided the best overall score. The results when using Trunk ($T_C$) sensor data slightly outperformed the scores when using Leg ($L_C$) sensor data.

The scores from LSTMs (M5 and M6) were moderate, but the issue was the very long runtime. The time for training LSTMs on raw signals was almost 5 times and features was almost 3 times of that for training the CNNs on visual features. All three of our ensemble architectures $M7$, $M8$ and $M9$, improved the scores of individual models. The majority voting model M9 had the best performance across all evaluation criteria without any extra training or parameter tuning. The scores were high for all evaluation criteria. The reported runtimes for ensemble models do not include the training time needed to prepare the constituent models.

Table 4.9 contains the scores of Ensemble architecture M7 with all features. The overall performance is vastly superior to using individual features. The performance is not affected much when the window size changes. A window size of 3 seconds generated the best scores for Ankle ($A_C$) and Leg ($L_C$) sensor data and a window size of 2 generated the best scores for data collected from the Trunk ($T_C$). Data from Leg ($L_C$) sensor provided the best overall scores. We can see that for M7, the runtimes were very large with all window sizes, which is a disadvantage considering this does not include the training time for individual models that make up the ensemble architecture. Adding the runtimes for constituent models, a significant amount of time was needed to train this model architecture. Training time is significant in the sense that when we incorporate more data modalities and increase data volume this will lead to that much more time consumption.

Table 4.10 reports the scores of Ensemble architecture M8 with all features. This architecture is similar to M7, except it adds a Average layer and calculates the average of the prediction of all constituent models, whereas M7 concatenates the predictions and uses 2 Dense layers to reshape the output. The overall performance is superior to using individual features and comparable to the performance of M7. The performance does not change significantly

when the window size changes. For this model, window size of 3 seconds generated the best scores using Ankle ($A_C$) and Leg ($L_C$) sensor data and a window size of 2 generated the best scores for data collected from the Trunk ($T_C$). Data from Trunk ($T_C$) sensor provided the best overall scores. The runtimes for m8 were not that large when compared to M7, but the performances were comparable. Thus, adding the runtimes for constituent models, M8 was able to produce similar results while needing a lot less time for training.

Table 4.11 presents the scores of Majority voting architecture M9 with all features. This architecture is different from M7 and M8, there is no training for this method. The overall performance is similar to the performance of M7 and M8. The performance is not majorly affected when the window size changes. For this model, window size of 3 seconds generated the best scores using Ankle ($A_C$) sensor data and a window size of 2 generated the best scores for data collected from the Trunk ($T_C$) and Leg ($L_C$). Data from Ankle ($A_C$) sensor provided the best overall scores, very closely followed by Trunk ($T_C$) and Leg ($L_C$). However the main strength of this model lies with its runtime. As this model only outputs the majority result of its constituent models, there is no training time, it can generate the output in milliseconds. It can produce similar scores to M7 and M8 while not needing any extra training time.

The results were also compared with the performance of some state-of-the-art models on the same dataset, as shown in Table 4.2. Mazilu et al [73] compared feature learning approaches based on time and statistical domain with unsupervised learning approaches using principle component analysis for both FOG detection and prediction. Their average sensitivity, specificity and $F_\beta$ score are presented for only the FOG class with both supervised and unsupervised approaches. Our proposed approach in this work outperforms their results for the FOG class. Moreover, the result they presented is only for the FOG class, their method had lower scores when identifying the PreFOG class.

Gokul et al [33] presented a number of Machine Learning based techniques to detect FOG events and evaluate their performances with sensitivity and specificity, which are also presented in Table 4.2. They also experimented with multiple window sizes and achieved the best results with a window length of 4

seconds. Although their performance is higher than our proposed model, they solved a binary classification problem of only identifying the FOG event. Their work does not have a prediction component. They achieved the best results with Random Forest classifier, with a sensitivity score of 98.91 and a specificity score of 99.44. Our ensemble architecture results were very close to their scores, while being able to also predict the onset of a FOG event. However, compared to Gokul et al [33], one shortcoming of our method would be the large size of the trained models, which might pose a problem in deploying the models to wearable sensors.

## 4.7 Application of Trained Model on Data collected from APDM™ sensors

In coordination with A. T. Still University, 14 PD patients took part in an experiment where their gait was monitored and various sensor data was recorded using APDM™ wearable sensors. Using the existing models trained on Daphnet [11] data, I attempted to monitor the onset of FOG on this data.

### 4.7.1 Data collection and Processing

The data was provided by the Arizona School of Health Sciences, A.T. Still University. The study was supervised, with the prior knowledge that 7 out of the 14 patients being Freezers, identified as a score  0 on the New Freezing of Gait Questionnaire (NFOGQ) and the rest being non-Freezers, with a score of zero on the NFOGQ [82]. There were 2 different configurations of the Six-Minute Walk Test (6MWT), a common clinical assessment of walking endurance [129]: 50 and 100 feet. For both configurations, the duration of the study was fixed at 6 minutes. Each patient had to walk for 6 minutes continuously with a 180 degree turn after either a 50 or 100 feet walk respectively. The number of turns was less for the 100 feet configuration. For each of the 14 patients, 2 datasets were generated with one for each configuration. The data was cleaned and missing values were filled with zeros. Twenty-eight data files were generated with 14 containing Freezer data and 14 containing non-Freezer

data.

#### 4.7.1.1 Sensor types and locations

The sensors were placed in 6 locations on a patients body. The location of the sensors can be seen in Figure 4.11.

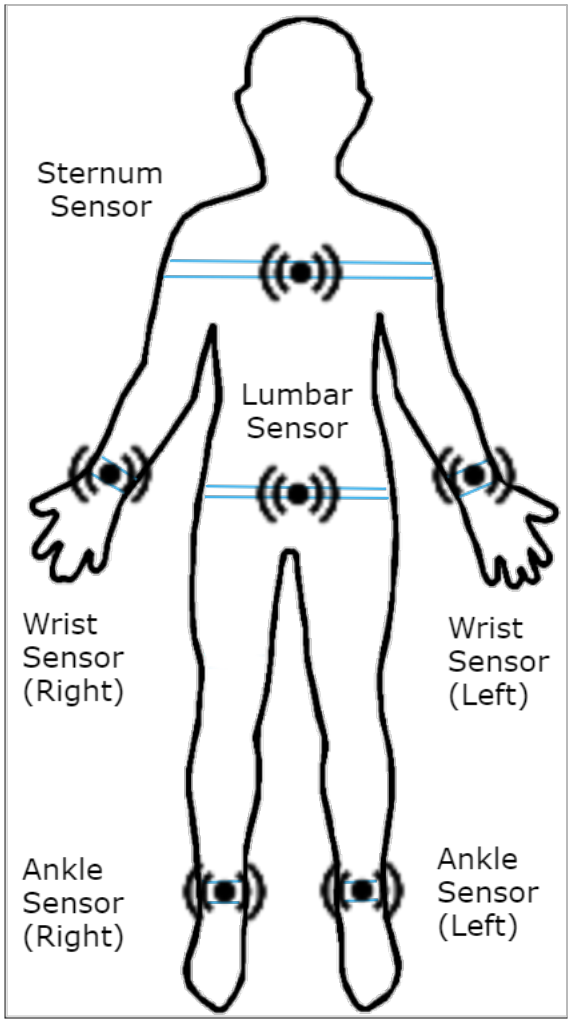



Figure 4.11: Sensor placement for data collection

The locations of the sensors are

- Left Foot (Ankle)
- Right Foot (Ankle)
- Left Wrist
- Right Wrist
- Sternum
- Lumber (Trunk)

For each location, there were 5 sensors recording data simultaneously. The sensors are as follows.

- Accelerometer
- Magnetometer
- Gyroscope
- Barometer
- Temperature

The recorded data was processed using Moveo Explorer™ and Mobility Lab™.

### 4.7.2 Workflow and Challenges

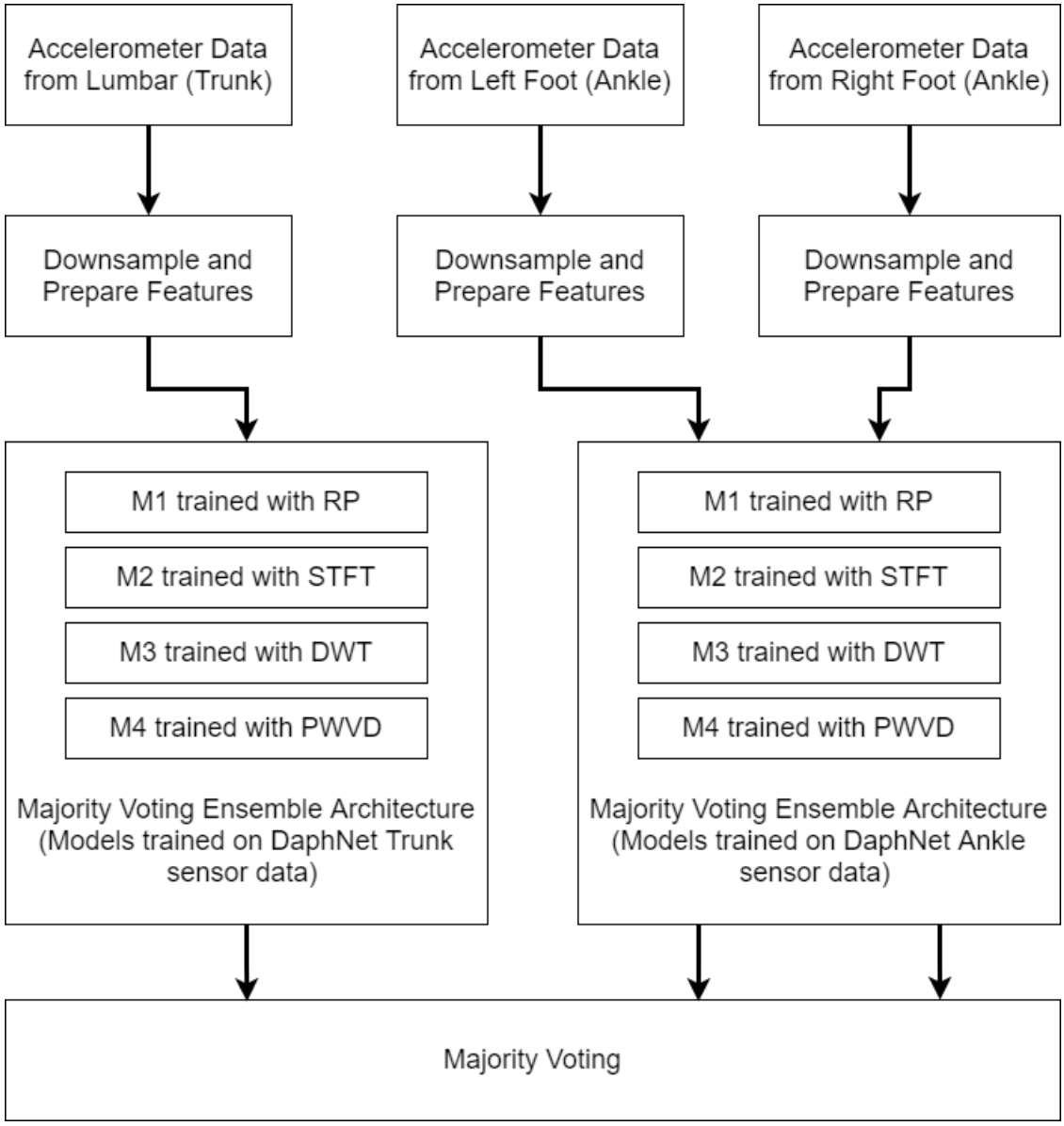



Figure 4.12: Workflow of monitoring FOG using models trained on Daphnet data

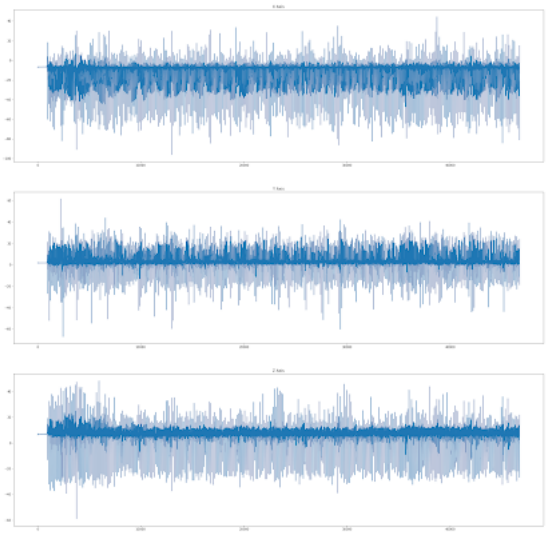
Figure 4.12 depicts our workflow for this part of the experiment. In order to make the data consistent across both Moveo Explorer and Mobility Lab, MinMax normalization was used. Since our purpose was only to monitor FOG, we discarded all the data where the patient did not experience FOG. That discarded data from 7 out of the 14 total patients.

As there were multiple sensors in multiple locations, finding the appropriate sensors for our experiment was an important step. Since there are multiple sensors producing a large amount of data, processing all of it it can be both time and resource consuming. Furthermore, it could impact the performance negatively. Identifying the optimum combination of sensors that were the most useful in detecting FOG posed a challenge. It was decided that we would only consider sensors locations that overlap with Daphnet [11] sensor locations, i.e. Left and Right foot (Ankle) and Lumber (Trunk). And although there were data from a varying number of sensors as well as derived kinematic information like velocities and displacements were available, we only considered accelerations. This was done because our training dataset, Daphnet [11] only provided acceleration data, so it was not possible for our existing architecture to use data from other sensors like Magnetometer, Gyroscope or Barometer. Our training dataset also did not consist of data from sensors placed on the wrist or sternum area of the patient, so signals from those areas would not be useful in this case. After that the signal from all three axis $(X, Y, Z)$ were combined using Equation 4.2.

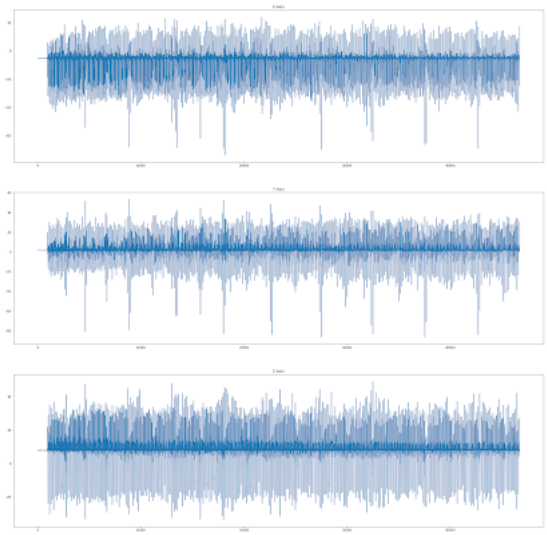
The sampling frequency for the data was 128 Hz, which is double the sampling frequency of Daphnet [11]. Since our models were trained on a sampling frequency of 64 Hz, the data needed to be down sampled from 128 Hz to 64 Hz in order for the trained models to be effective.

Although the data was labeled, the labels were for the whole time series. So one continuous 6 minute time series had only one label indicating whether FOG occurred in that series or not. The exact occurrence of FOG was unmarked. Thus even if we did apply our models trained on Daphnet [11], where the exact time instance of FOG was marked on this data, it would be difficult to verify the accuracy of the results. Figure 4.13 shows some accelerometer signals from both Freezer and non Freezer data. No significant difference can be visually defined between the two, because the FOG episodes occur in between regular walking. So even the majority of Freezer data is basically similar to the non Freezer data, with some FOG episodes in between.

Figure 4.13: Comparison of Freezer vs Non Freezer Accelerometer signals



(a) Accelerometer signal representing Freezer data



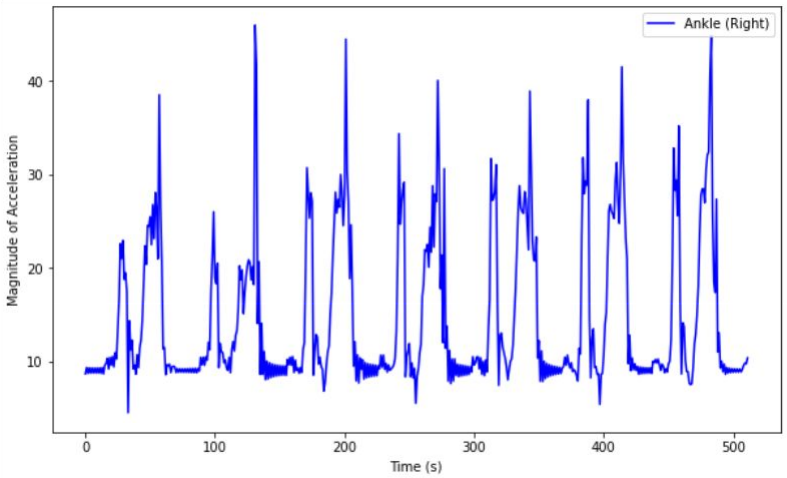
(b) Accelerometer signal representing non Freezer data

For detecting even a very small occurrence of FOG, a window size of 2 seconds was selected. After selecting the appropriate sensors and downsam-

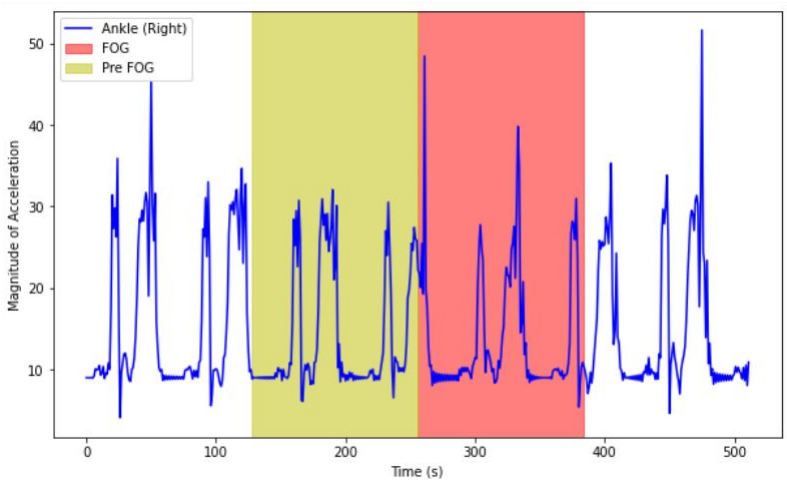
pling the data, non overlapping moving windows of 2 seconds were extracted from the source signal. Then the relevant features were generated from the windows. We only chose to use visual features $RP, STFT, DWT, PWVD$, since they showed superior performance and were less time consuming during the previous part of our experiment.

After generating the features, the models M1, M2, M3 and M4 trained on Daphnet [11] data were used to analyze the data. There were three instances of trained models. For data from Left and Right foot, the instances of models trained on data from Ankle sensor of Daphnet was used, and for data from Lumbar, models trained on data from Trunk sensor of Daphnet was used. Our training dataset Daphnet does not specify whether the sensors were located at the left of right side of the body. So Left and Right foot data was used with models trained on Ankle sensor. Finally the results for each of the models were passed through a majority voting model, providing us with a result.

Figure 4.14: Comparison of Freezer vs Non Freezer Accelerometer signals, with detected FOG, PreFOG and Non FOG regions highlighted




(a) Accelerometer signal representing Non Freezer data




(b) Accelerometer signal representing Freezer data with detected Pre FOG(Yellow) and FOG(Red) region

Figure 4.14 shows a small 4 windows chunk of the signal from Accelerometer signal from right Ankle with a window size of 2 seconds for comparison. Since our data was downsampled, this signal has a sampling rate of 64 Hz instead of 128 Hz. The first figure shows a part of the signal that does not contain any FOG or PreFOG according to our model. When we compare it with the marked PreFOG and FOG region in the second figure, we can visually identify some differences. For the PreFOG region, all the peak heights are smaller than that of NonFOG signal. The FOG region has higher peaks than that of PreFOG, but the average peak height still appear to be lower than

that of Non FOG signals. These patterns appear to be consistent with our observations from Daphnet Data presented in Figure 4.3.

## 4.8 Conclusions

In this work, the performance of multiple time frequency representation techniques were compared in detecting and predicting FOG using tri axial accelerometer sensor data from the publicly available Daphnet [11] dataset. Three ensemble neural network architectures comprised of multiple modalities of data were proposed and their performance was analyzed. It was established that ensemble network architectures significantly improve the performance over individual models. I applied some of the trained models to monitor the progression of FOG from accelerometer data captured using APDM wearable sensors. In future works, I would like to integrate more data modalities, improve the model selection process for creating ensemble architectures, reduce size and complexity of the models and finally apply the resultant models to more real-world data. More importantly, I will transfer our models in identifying the PreFOG class i.e., predicting FOG events, to use real-world data from wearable sensors, in order to test the potential of preventing falls by initiating RAS even before the start of the event. One of the drawbacks of proposed system was that the performance was not verified when applied on collected data due to lack of proper annotations. Thus, future goals also include properly testing the performance of this method with collected data using wearable sensors and develop it further so that the system is capable of administering RAS upon prediction of FOG.

# Chapter 5

# Conclusion And Future Directions

In this work, automatic CAD based systems are proposed for the diagnosis and monitoring of PD. Multiple contributions were made using 2 kinds of data, namely Neuroimaging Data (MRI) and motion data captured using accelerometer sensors.

For the first part, a non-invasive method for PD detection is proposed using T1-weighted MRI scans of the human brain. The proposed solution is transfer learning based, using models trained on unrelated ImageNet [20] dataset, and does not require a large number of training data. Models designed for the ILSVRC were used for this problem, both individually and in multiple ensemble structures. It was discovered that instead of using the Whole Brain scans, extracting the GM and WM yields better performance. Multiple evaluation criteria was used to properly validate the performance of the proposed solution.

Occlusion analysis was also performed on models that achieved notable performance in this task, to identify regions of interest from the extracted GM and WM scans. Since the system is fully automated and does not require human supervision, it can play a vital role in remove human errors and biases in PD detection as well as reduce detection time and save resources. For future works, the relevant areas identified during occlusion analysis need to be focused on. Focusing on a small region of the brain instead of the whole brain scan would reduce the volume of the data needed to be processed, lowering

the usage of both time and hardware resources.

In the second part of the thesis, the performance of various time frequency representation techniques to predict and detect FOG was analyzed. 2 NN based architectures were proposed and 3 ensemble architectures were designed combining various data modalities with the proposed NN architectures. The performance of the models were evaluated on the publicly available Daphnet [11] dataset. The trained models were also used to monitor the progression of FOG from a dataset collected by A. T. Still University. Detecting the onset of FOG can play a vital role in helping the patients lead a healthy lifestyle. Early detection can help in starting RAS and helping patients resume normal gait.

There are multiple possibilities for future research. For both PD detection and FOG monitoring, improving the model performance is vital. Furthermore, deploying the proposed systems to be used in real world scenarios and evaluating their performance remains a challenge.

# References


[1] [Online]. Available: https://www.movementdisorders.org/MDS.htm (visited on 06/18/2021). 2, 3

[2] [Online]. Available: https://www.fil.ion.ucl.ac.uk/spm/doc/spm12_manual.pdf (visited on 06/18/2021). 24

[3] O. Aevarsson, A. Svanborg, and I. Skoog, "Seven-year survival rate after age 85 years: Relation to alzheimer disease and vascular dementia," *Archives of Neurology*, vol. 55, no. 9, pp. 1226–1232, 1998. 3

[4] M. Agrawal and A. Biswas, "Molecular diagnostics of neurodegenerative disorders," *Frontiers in Molecular Biosciences*, vol. 2, 2015. DOI: 10.3389/fmolb.2015.00054. 2

[5] C. Ahlrichs, A. Samá, M. Lawo, *et al.*, "Detecting freezing of gait with a tri-axial accelerometer in parkinson's disease patients," *Medical & Biological Engineering & Computing*, vol. 54, no. 1, pp. 223–233, 2015. DOI: 10.1007/s11517-015-1395-3. 6

[6] N. E. Allen, A. K. Schwarzel, and C. G. Canning, "Recurrent falls in parkinson's disease: A systematic review," *Parkinson's Disease*, vol. 2013, pp. 1–16, 2013. DOI: 10.1155/2013/906274. 5

[7] P. Arias and J. Cudeiro, "Effect of rhythmic auditory stimulation on gait in parkinsonian patients with and without freezing of gait," *PLoS ONE*, vol. 5, no. 3, 2010. DOI: 10.1371/journal.pone.0009675. 7

[8] T. Ashfaque Mostafa and I. Cheng, "Parkinson's disease detection using ensemble architecture from mr images*," *2020 IEEE 20th International Conference on Bioinformatics and Bioengineering (BIBE)*, 2020. DOI: 10.1109/bibe50027.2020.00167. 8

[9] D. Athauda, K. Maclagan, S. S. Skene, *et al.*, "Exenatide once weekly versus placebo in parkinson's disease: A randomised, double-blind, placebo-controlled trial," *The Lancet*, vol. 390, no. 10103, pp. 1664–1675, 2017. DOI: 10.1016/s0140-6736(17)31585-4. 4

[10] M. Bachlin, M. Plotnik, D. Roggen, *et al.*, "Wearable assistant for parkinson's disease patients with the freezing of gait symptom," *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 2, pp. 436–446, Mar. 2010. DOI: 10.1109/titb.2009.2036165. [Online]. Available: https://doi.org/10.1109/titb.2009.2036165. 14

[11] M. Bachlin, M. Plotnik, D. Roggen, *et al.*, "Wearable assistant for parkinson's disease patients with the freezing of gait symptom," *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 2, pp. 436–446, 2010. DOI: 10.1109/TITB.2009.2036165. 5, 7, 14, 15, 17, 69, 106,

[12] H. Braak, K. D. Tredici, U. Rüb, R. A. de Vos, E. N. Jansen Steur, and E. Braak, "Staging of brain pathology related to sporadic parkinson's disease," *Neurobiology of Aging*, vol. 24, no. 2, pp. 197–211, 2003. DOI: 10.1016/s0197-4580(02)00065-9. 3, 12

[13] M. Buda, A. Maki, and M. A. Mazurowski, "A systematic study of the class imbalance problem in convolutional neural networks," *Neural Networks*, vol. 106, pp. 249–259, 2018. DOI: 10.1016/j.neunet.2018.07.011. 12

[14] J. Camps, A. Samà, M. Martín, *et al.*, in *Advances in Computational Intelligence*, I. Rojas, G. Joya, and A. Catala, Eds., Cham: Springer International Publishing, 2017, pp. 344–355, ISBN: 978-3-319-59147-6. 14

[15] J. Camps, A. Samà, M. Martín, *et al.*, "Deep learning for freezing of gait detection in parkinson's disease patients in their homes using a waist-worn inertial measurement unit," *Knowledge-Based Systems*, vol. 139, pp. 119–131, 2018. DOI: 10.1016/j.knosys.2017.10.017. 17

[16] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002. DOI: 10.1613/jair.953. 76

[17] H.-C. Cheng, C. M. Ulane, and R. E. Burke, "Clinical progression in parkinson disease and the neurobiology of axons," *Annals of Neurology*, vol. 67, no. 6, pp. 715–725, 2010. DOI: 10.1002/ana.21995. 3

[18] H. Choi, S. Ha, H. J. Im, S. H. Paek, and D. S. Lee, "Refining diagnosis of parkinson's disease with deep learning-based interpretation of dopamine transporter imaging," *NeuroImage: Clinical*, vol. 16, pp. 586–594, 2017. DOI: 10.1016/j.nicl.2017.09.010. 11

[19] J. H. Cole, R. E. Marioni, S. E. Harris, and I. J. Deary, "Brain age and other bodily 'ages': Implications for neuropsychiatry," *Molecular Psychiatry*, vol. 24, no. 2, pp. 266–281, 2018. DOI: 10.1038/s41380-018-0098-1. 22–24

[20] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009. DOI: 10.1109/cvprw.2009.5206848. 9, 12, 13, 21, 57, 67, 115

[21] "Design and implementation of the wordnet lexical database and searching software," *WordNet*, 1998. DOI: 10.7551/mitpress/7287.003.0009. 12, 13

[22] J.-P. Eckmann, S. O. Kamphorst, and D. Ruelle, "Recurrence plots of dynamical systems," *Europhysics Letters (EPL)*, vol. 4, no. 9, pp. 973–977, 1987. DOI: 10.1209/0295-5075/4/9/004. 79, 81

[23] P. Elizabeth Boskey, *Learn about person years and person months in research studies*. [Online]. Available: https://www.verywellhealth.com/person-years-and-person-months-3132812. 2

[24] S. Esmaeilzadeh, Y. Yang, and E. Adeli, *End-to-end parkinson disease diagnosis using brain mr-images by 3d-cnn*, 2018. arXiv: 1806.05233 [cs.CV]. 11

[25] S. Fahn, "Classification of movement disorders," *Movement Disorders*, vol. 26, no. 6, pp. 947–957, 2011. DOI: 10.1002/mds.23759. 1

[26] M. L. Ferster, S. Mazilu, and G. Tröster, "Gait parameters change prior to freezing in parkinson's disease: A data-driven study with wearable inertial units," *Proceedings of the 10th EAI International Conference on Body Area Networks*, 2015. DOI: 10.4108/eai.28-9-2015.2261411. 5, 6

[27] N. K. Focke, G. Helms, S. Scheewe, *et al.*, "Individual voxel-based subtype prediction can differentiate progressive supranuclear palsy from idiopathic parkinson syndrome and healthy controls," *Human Brain Mapping*, vol. 32, no. 11, pp. 1905–1915, 2011. DOI: 10.1002/hbm.21161. 11, 36

[28] N. Garcia, *Movement disorders*. [Online]. Available: https://www.neuromodulation.com/movement-disorders (visited on 06/16/2021). 1

[29] N. Giladi, H. Shabtai, E. Simon, S. Biran, J. Tal, and A. Korczyn, "Construction of freezing of gait questionnaire for patients with parkinsonism," *Parkinsonism & Related Disorders*, vol. 6, no. 3, pp. 165–170, 2000. DOI: 10.1016/s1353-8020(99)00062-0. 6

[30] N. Giladi, T. A. Treves, E. S. Simon, *et al.*, "Freezing of gait in patients with advanced parkinson's disease," *Journal of Neural Transmission*, vol. 108, no. 1, pp. 53–61, 2001. DOI: 10.1007/s007020170096. 5

[31] N. Giladi, J. Tal, T. Azulay, *et al.*, "Validation of the freezing of gait questionnaire in patients with parkinson's disease," *Movement Disorders*, vol. 24, no. 5, pp. 655–661, 2009. DOI: 10.1002/mds.21745. 6

[32] C. G. Goetz, *The history of parkinson's disease: Early clinical descriptions and neurological therapies*, Jan. 1970. [Online]. Available: http://perspectivesinmedicine.cshlp.org/lookup/doi/10.1101/cshperspect.a008862. 3

[33] H. Gokul, P. Suresh, B. Hari Vignesh, R. Pravin Kumaar, and V. Vijayaraghavan, "Gait recovery system for parkinson's disease using machine learning on embedded platforms," *2020 IEEE International Systems Conference (SysCon)*, 2020. DOI: 10.1109/syscon47679.2020.9275930. 92, 105, 106

[34] J. Gorodkin, "Comparing two k-category assignments by a k-category correlation coefficient," *Computational Biology and Chemistry*, vol. 28, no. 5-6, pp. 367–374, 2004. DOI: 10.1016/j.compbiolchem.2004.09.006. 20

[35] A. Haar, "Zur theorie der orthogonalen funktionensysteme," *Mathematische Annalen*, vol. 71, no. 1, pp. 38–53, 1911. DOI: 10.1007/bf01456927. 82

[36] J. H. Han, W. J. Lee, T. B. Ahn, B. S. Jeon, and K. S. Park, "Gait analysis for freezing detection in patients with movement disorder using three dimensional acceleration system," in *Proceedings of the 25th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (IEEE Cat. No.03CH37439)*, IEEE. DOI: 10.1109/iembs.2003.1279781. [Online]. Available: https://doi.org/10.1109/iembs.2003.1279781. 14

[37] L. Hansen and P. Salamon, "Neural network ensembles," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 10, pp. 993–1001, 1990. DOI: 10.1109/34.58871. 8

[38] T. Hashimoto, "Speculation on the responsible sites and pathophysiology of freezing of gait," *Parkinsonism & Related Disorders*, vol. 12, 2006. DOI: 10.1016/j.parkreldis.2006.05.017. 6

[39] M. M. Hassan, S. Huda, M. Z. Uddin, A. Almogren, and M. Alrubaian, "Human activity recognition from body sensor data using deep learning," *Journal of Medical Systems*, vol. 42, no. 6, 2018. DOI: 10.1007/s10916-018-0948-z. 7, 8

[40] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. DOI: 10.1109/cvpr.2016.90. 30, 53, 57–60

[41] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997. DOI: 10.1162/neco.1997.9.8.1735. 8

[42] M. Hoehn and M. Yahr, "Parkinsonism: Onset, progression, and mortality," *Neurology*, vol. 77, no. 9, pp. 874–874, 2011. DOI: 10.1212/01.wnl.0000405146.06300.91. 69

[43] M. Hosseinzadeh, "Robust control applications in biomedical engineering: Control of depth of hypnosis," *Control Applications for Biomedical Engineering Systems*, pp. 89–125, 2020. DOI: 10.1016/b978-0-12-817461-6.00004-4. 82

[44] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, *Densely connected convolutional networks*, 2018. arXiv: 1608.06993 [cs.CV]. 30, 53

[45] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, *Squeezenet: Alexnet-level accuracy with 50x fewer parameters and ¡0.5mb model size*, 2016. arXiv: 1602.07360 [cs.CV]. 30, 54

[46] *Ixi dataset*. [Online]. Available: https://brain-development.org/ixi-dataset/ (visited on 06/18/2021). 22

[47] C. Janiesch, P. Zschech, and K. Heinrich, "Machine learning and deep learning," *Electronic Markets*, Apr. 2021. DOI: 10.1007/s12525-021-00475-2. [Online]. Available: https://doi.org/10.1007/s12525-021-00475-2. 7

[48] M. JENKINSON, "Bet2 : Mr-based estimation of brain, skull and scalp surfaces," *Eleventh Annual Meeting of the Organization for Human Brain Mapping, 2005*, 2005. [Online]. Available: https://ci.nii.ac.jp/naid/10030066593/en/. 25

[49] M. Jenkinson, P. Bannister, M. Brady, and S. Smith, "Improved optimization for the robust and accurate linear registration and motion correction of brain images," *NeuroImage*, vol. 17, no. 2, pp. 825–841, 2002. DOI: 10.1006/nimg.2002.1132. 25

[50] M. Jenkinson and S. Smith, "A global optimisation method for robust affine registration of brain images," *Medical Image Analysis*, vol. 5, no. 2, pp. 143–156, 2001. DOI: 10.1016/s1361-8415(01)00036-6. 25

[51] S. B. Johnson, R. W. Blum, and J. N. Giedd, "Adolescent maturity and the brain: The promise and pitfalls of neuroscience research in adolescent health policy," *Journal of Adolescent Health*, vol. 45, no. 3, pp. 216–221, 2009. DOI: 10.1016/j.jadohealth.2009.05.016. 22

[52] E. Jovanov, E. Wang, L. Verhagen, M. Fredrickson, and R. Fratangelo, "A real time system for detection and unfreezing of gait of parkinson's patients," *2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2009. DOI: 10.1109/iembs.2009.5334257. 7

[53] L. Kalilani, M. Asgharnejad, T. Palokangas, and T. Durgin, "Comparing the incidence of falls/fractures in parkinson's disease patients in the us population," *PLOS ONE*, vol. 11, no. 9, 2016. DOI: 10.1371/journal.pone.0161689. 2, 5

[54] F. Karim, S. Majumdar, H. Darabi, and S. Chen, "Lstm fully convolutional networks for time series classification," *IEEE Access*, vol. 6, pp. 1662–1669, 2018. DOI: 10.1109/ACCESS.2017.2779939. 8

[55] G. K. Kerr, C. J. Worringham, M. H. Cole, P. F. Lacherez, J. M. Wood, and P. A. Silburn, "Predictors of future falls in parkinson disease," *Neurology*, vol. 75, no. 2, pp. 116–124, 2010. DOI: 10.1212/wnl.0b013e3181e7b688. 6

[56] H. Kim, H. J. Lee, W. Lee, *et al.*, "Unconstrained detection of freezing of gait in parkinson's disease patients using smartphone," *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2015. DOI: 10.1109/embc.2015.7319209. 6

[57] W. Koller, R. O'Hara, W. Weiner, *et al.*, "Relationship of aging to parkinson's disease.," *Advances in neurology*, vol. 45, pp. 317–321, 1987. 2

[58] S. Kornblith, J. Shlens, and Q. V. Le, "Do better imagenet models transfer better?" *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. DOI: 10.1109/cvpr.2019.00277. 13

[59] N. Kubis, B. A. Faucheux, G. Ransmayr, *et al.*, "Preservation of midbrain catecholaminergic neurons in very old human subjects," *Brain*, vol. 123, no. 2, pp. 366–373, 2000. DOI: 10.1093/brain/123.2.366. 2

[60] K. J. Kubota, J. A. Chen, and M. A. Little, "Machine learning for large-scale wearable sensor data in parkinson's disease: Concepts, promises, pitfalls, and futures," *Movement Disorders*, vol. 31, no. 9, pp. 1314–1326, 2016. DOI: 10.1002/mds.26693. 7

[61] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1192–1209, 2013. DOI: 10.1109/surv.2012.110112.00192. 7

[62] M. D. Latt, S. R. Lord, J. G. Morris, and V. S. Fung, "Clinical and physiological assessments for elucidating falls risk in parkinson's disease," *Movement Disorders*, vol. 24, no. 9, pp. 1280–1289, 2009. DOI: 10.1002/mds.22561. 6

[63] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech, and time series," in *The Handbook of Brain Theory and Neural Networks*. Cambridge, MA, USA: MIT Press, 1998, pp. 255–258, ISBN: 0262511029. 8

[64] I. Lim, E. van Wegen, C. de Goede, *et al.*, "Effects of external rhythmical cueing on gait in patients with parkinson's disease: A systematic review," *Clinical Rehabilitation*, vol. 19, no. 7, pp. 695–713, 2005. DOI: 10.1191/0269215505cr906oa. 7

[65] M. Little, P. McSharry, E. Hunter, J. Spielman, and L. Ramig, "Suitability of dysphonia measurements for telemonitoring of parkinson's disease," *Nature Precedings*, 2008. DOI: 10.1038/npre.2008.2298.1. 4

[66] D. Long, J. Wang, M. Xuan, *et al.*, "Automatic classification of early parkinson's disease with multi-modal mr imaging," *PLoS ONE*, vol. 7, no. 11, 2012. DOI: 10.1371/journal.pone.0047714. 12

[67] E. D. Louis and J. J. Ferreira, "How common is the most common adult movement disorder? update on the worldwide prevalence of essential tremor," *Movement Disorders*, vol. 25, no. 5, pp. 534–541, 2010. DOI: 10.1002/mds.22838. 3

[68] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, *Shufflenet v2: Practical guidelines for efficient cnn architecture design*, 2018. arXiv: 1807.11164 [cs.CV]. 30, 54

[69] W. Maetzler, J. Domingos, K. Srulijes, J. J. Ferreira, and B. R. Bloem, "Quantitative wearable sensors for objective assessment of parkinson's disease," *Movement Disorders*, vol. 28, no. 12, pp. 1628–1637, 2013. DOI: 10.1002/mds.25628. 7

[70] S. Marcel and Y. Rodriguez, "Torchvision the machine-vision package of torch," *Proceedings of the international conference on Multimedia - MM '10*, 2010. DOI: 10.1145/1873951.1874254. 30

[71] R. Matias, V. Paixão, R. Bouça, and J. J. Ferreira, "A perspective on wearable sensor measurements and data science for parkinson's disease," *Frontiers in Neurology*, vol. 8, 2017. DOI: 10.3389/fneur.2017.00677. 7

[72] B. Matthews, "Comparison of the predicted and observed secondary structure of t4 phage lysozyme," *Biochimica et Biophysica Acta (BBA) - Protein Structure*, vol. 405, no. 2, pp. 442–451, 1975. DOI: 10.1016/0005-2795(75)90109-9. 20

[73] S. Mazilu, A. Calatroni, E. Gazit, D. Roggen, J. M. Hausdorff, and G. Tröster, "Feature learning for detection and prediction of freezing of gait in parkinson's disease," *Machine Learning and Data Mining in Pattern Recognition*, pp. 144–158, 2013. DOI: 10.1007/978-3-642-39712-7_11. 74, 92, 105

[74] S. Mazilu, M. Hardegger, Z. Zhu, *et al.*, "Online detection of freezing of gait with smartphones and machine learning techniques," in *Proceedings of the 6th International Conference on Pervasive Computing Technologies for Healthcare*, IEEE, 2012. DOI: 10.4108/icst.pervasivehealth.2012.248680. [Online]. Available: https://doi.org/10.4108/icst.pervasivehealth.2012.248680. 15, 17, 18

[75] *Metrics and scoring: Quantifying the quality of predictions*. [Online]. Available: https://scikit-learn.org/stable/modules/model_evaluation.html#matthews-corrcoef. 20

[76] P. P. Michel, E. C. Hirsch, and S. Hunot, "Understanding dopaminergic cell death pathways in parkinson disease," *Neuron*, vol. 90, no. 4, pp. 675–691, 2016. DOI: 10.1016/j.neuron.2016.03.038. 3

[77] O. Moore, C. Peretz, and N. Giladi, "Freezing of gait affects quality of life of peoples with parkinson's disease beyond its relationships with mobility and gait," *Movement Disorders*, vol. 22, no. 15, pp. 2192–2195, 2007. DOI: 10.1002/mds.21659. 6

[78] S. T. Moore, D. A. Yungher, T. R. Morris, *et al.*, "Autonomous identification of freezing of gait in parkinson's disease from lower-body segmental accelerometry," *Journal of NeuroEngineering and Rehabilitation*, vol. 10, no. 1, p. 19, 2013. DOI: 10.1186/1743-0003-10-19. [Online]. Available: https://doi.org/10.1186/1743-0003-10-19. 15

[79] S. T. Moore, H. G. MacDougall, and W. G. Ondo, "Ambulatory monitoring of freezing of gait in parkinson's disease," *Journal of Neuroscience Methods*, vol. 167, no. 2, pp. 340–348, 2008. DOI: 10.1016/j.jneumeth.2007.08.023. 6

[80] ——, "Ambulatory monitoring of freezing of gait in parkinson's disease," *Journal of Neuroscience Methods*, vol. 167, no. 2, pp. 340–348, Jan. 2008. DOI: 10.1016/j.jneumeth.2007.08.023. [Online]. Available: https://doi.org/10.1016/j.jneumeth.2007.08.023. 14

[81] T. A. Mostafa and I. Cheng, *Parkinson's disease detection with ensemble architectures based on ilsvrc models*, 2020. arXiv: 2007.12496 [eess.IV]. 8

[82] A. Nieuwboer, L. Rochester, T. Herman, *et al.*, "Reliability of the new freezing of gait questionnaire: Agreement between patients with parkinson's disease and their carers," *Gait & Posture*, vol. 30, no. 4, pp. 459–463, 2009. DOI: 10.1016/j.gaitpost.2009.07.108. 6, 106

[83] A. Nieuwboer, W. d. Weerdt, R. Dom, and E. Lesaffre, "A frequency and correlation analysis of motor deficits in parkinson patients," *Disability and Rehabilitation*, vol. 20, no. 4, pp. 142–150, 1998. DOI: 10.3109/09638289809166074. 6

[84] J. G. Nutt, B. R. Bloem, N. Giladi, M. Hallett, F. B. Horak, and A. Nieuwboer, "Freezing of gait: Moving forward on a mysterious clinical phenomenon," *The Lancet Neurology*, vol. 10, no. 8, pp. 734–744, 2011. DOI: 10.1016/s1474-4422(11)70143-0. 5

[85] H. F. Nweke, Y. W. Teh, M. A. Al-garadi, and U. R. Alo, "Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: State of the art and research challenges," *Expert Systems with Applications*, vol. 105, pp. 233–261, 2018. DOI: 10.1016/j.eswa.2018.03.056. 7, 8

[86] Y. Okuma, "Freezing of gait and falls in parkinson's disease," *Journal of Parkinson's Disease*, vol. 4, no. 2, pp. 255–260, 2014. DOI: 10.3233/jpd-130282. 5, 6

[87] Y. Okuma and N. Yanagisawa, "The clinical spectrum of freezing of gait in parkinson's disease," *Movement Disorders*, vol. 23, no. S2, 2008. DOI: 10.1002/mds.21934. 5

[88] F. Ordóñez and D. Roggen, "Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition," *Sensors*, vol. 16, no. 1, p. 115, 2016. DOI: 10.3390/s16010115. 7, 8

[89] N. K. Orphanidou, A. Hussain, R. Keight, P. Lishoa, J. Hind, and H. Al-Askar, "Predicting freezing of gait in parkinsons disease patients using machine learning," in *2018 IEEE Congress on Evolutionary Computation (CEC)*, IEEE, Jul. 2018. DOI: 10.1109/cec.2018.8477909. [Online]. Available: https://doi.org/10.1109/cec.2018.8477909. 16

[90] ——, "Predicting freezing of gait in parkinsons disease patients using machine learning," in *2018 IEEE Congress on Evolutionary Computation (CEC)*, 2018, pp. 1–8. DOI: 10.1109/CEC.2018.8477909. 16

[91] F. L. Pagan, "Improving outcomes through early diagnosis of Parkinson's disease," *Am J Manag Care*, vol. 18, no. 7 Suppl, S176–182, Sep. 2012. ii, 4, 5

[92] R. Pascanu, C. Gulcehre, K. Cho, and Y. Bengio, *How to construct deep recurrent neural networks*, 2014. arXiv: 1312.6026 [cs.NE]. 8

[93] A. Paszke, S. Gross, F. Massa, *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32, Curran Associates, Inc., 2019. [Online]. Available: https://proceedings.neurips.cc/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf. 30

[94] S. Pazhanirajan and D. P. Dhanalakshmi, "Classification of parkinson's disease using mri images," 2016. 11

[95] B. Peng, S. Wang, Z. Zhou, *et al.*, "A multilevel-roi-features-based machine learning method for detection of morphometric biomarkers in parkinson's disease," *Neuroscience Letters*, vol. 651, pp. 88–94, 2017. DOI: 10.1016/j.neulet.2017.04.034. 3, 11

[96] C. R. Pereira, D. R. Pereira, G. H. Rosa, *et al.*, "Handwritten dynamics assessment through convolutional neural networks: An application to parkinson's disease identification," *Artificial Intelligence in Medicine*, vol. 87, pp. 67–77, 2018. DOI: 10.1016/j.artmed.2018.04.001. 4

[97] *Population ages 65 and above*. [Online]. Available: https://data.worldbank.org/indicator/SP.POP.65UP.TO.ZS (visited on 06/21/2021). 2

[98] B. Rajoub, "Characterization of biomedical signals: Feature engineering and extraction," *Biomedical Signal Processing and Artificial Intelligence in Healthcare*, pp. 29–50, 2020. DOI: 10.1016/b978-0-12-818946-7.00002-0. 81

[99] B. Rana, A. Juneja, M. Saxena, *et al.*, "Graph-theory-based spectral feature selection for computer aided diagnosis of parkinson's disease using t1-weighted mri," *International Journal of Imaging Systems and Technology*, vol. 25, no. 3, pp. 245–255, 2015. DOI: 10.1002/ima.22141. 12, 36

[100] B. Rana, A. Juneja, M. Saxena, *et al.*, "Regions-of-interest based automated diagnosis of parkinson's disease using t1-weighted mri," *Expert Systems with Applications*, vol. 42, no. 9, pp. 4506–4516, 2015. DOI: 10.1016/j.eswa.2015.01.062. 4

[101] B. Rana, A. Juneja, M. Saxena, *et al.*, "Relevant 3d local binary pattern based features from fused feature descriptor for differential diagnosis of parkinson's disease using structural mri," *Biomedical Signal Processing and Control*, vol. 34, pp. 134–143, 2017. DOI: 10.1016/j.bspc.2017.01.007. 12, 21, 36

[102] *Recurrence plot*, Apr. 2021. [Online]. Available: https://en.wikipedia.org/wiki/Recurrence_plot. 79

[103] A. Reeve, E. Simcox, and D. Turnbull, "Ageing and parkinson's disease: Why is advancing age the biggest risk factor?" *Ageing Research Reviews*, vol. 14, pp. 19–30, 2014, ISSN: 1568-1637. DOI: https://doi.org/10.1016/j.arr.2014.01.004. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1568163714000051. 2, 3

[104] J. Rieke, F. Eitel, M. Weygandt, J.-D. Haynes, and K. Ritter, "Visualizing convolutional networks for mri-based diagnosis of alzheimer's disease," *Understanding and Interpreting Machine Learning in Medical Image Computing Applications*, pp. 24–31, 2018. DOI: 10.1007/978-3-030-02628-8_3. 57

[105] P. A. Rocha, G. M. Porfírio, H. B. Ferraz, and V. F. Trevisani, "Effects of external cues on gait parameters of parkinson's disease patients: A systematic review," *Clinical Neurology and Neurosurgery*, vol. 124, pp. 127–134, 2014. DOI: 10.1016/j.clineuro.2014.06.026. 7

[106] D. Rodríguez-Martín, A. Samà, C. Pérez-López, *et al.*, "Home detection of freezing of gait using support vector machines through a single waist-worn triaxial accelerometer," *PLOS ONE*, vol. 12, no. 2, 2017. DOI: 10.1371/journal.pone.0171764. 6, 16, 18

[107] O. Russakovsky, J. Deng, H. Su, *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015. DOI: 10.1007/s11263-015-0816-y. 12, 21, 30

[108] I. Rustempasic and M. Can, "Diagnosis of parkinson's disease using fuzzy c-means clustering and pattern recognition," *Southeast Europe Journal of Soft Computing*, vol. 2, no. 1, 2013. DOI: 10.21533/scjournal.v2i1.44. 4

[109] A. Sakalauskas, A. Lukoševičius, K. Laučkaitė, D. Jegelevičius, and S. Rutkauskas, "Automated segmentation of transcranial sonographic images in the diagnostics of parkinson's disease," *Ultrasonics*, vol. 53, no. 1, pp. 111–121, 2013. DOI: 10.1016/j.ultras.2012.04.005. 4

[110] C. Salvatore, A. Cerasa, I. Castiglioni, *et al.*, "Machine learning on brain mri data for differential diagnosis of parkinson's disease and progressive supranuclear palsy," *Journal of Neuroscience Methods*, vol. 222, pp. 230–237, 2014. DOI: 10.1016/j.jneumeth.2013.11.016. 11

[111] A. Samà, D. Rodríguez-Martín, C. Pérez-López, *et al.*, "Determining the optimal features in freezing of gait detection through a single waist accelerometer in home environments," *Pattern Recognition Letters*, vol. 105, pp. 135–143, 2018. DOI: 10.1016/j.patrec.2017.05.009. 16

[112] R. San-Segundo, J. M. Montero, R. Barra-Chicote, F. Fernández, and J. M. Pardo, "Feature extraction from smartphone inertial signals for human activity segmentation," *Signal Processing*, vol. 120, pp. 359–372, 2016, ISSN: 0165-1684. DOI: https://doi.org/10.1016/j.sigpro.2015.09.029. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S016516841500331X. 17, 18

[113] R. San-Segundo, H. Navarro-Hellín, R. Torres-Sánchez, J. Hodgins, and F. De la Torre, "Increasing robustness in the detection of freezing of gait in parkinson's disease," *Electronics*, vol. 8, no. 2, 2019, ISSN: 2079-9292. DOI: 10.3390/electronics8020119. [Online]. Available: https://www.mdpi.com/2079-9292/8/2/119. 17

[114] Á. Sánchez-Ferro, M. Elshehabi, C. Godinho, *et al.*, "New methods for the assessment of parkinson's disease (2005 to 2015): A systematic review," *Movement Disorders*, vol. 31, no. 9, pp. 1283–1292, 2016. DOI: 10.1002/mds.26723. 7

[115] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, *Mobilenetv2: Inverted residuals and linear bottlenecks*, 2019. arXiv: 1801.04381 [cs.CV]. 30, 54

[116] G. Sateesh Babu, S. Suresh, and B. S. Mahanand, "A novel pbl-mcrbfn-rfe approach for identification of critical brain regions responsible for parkinson's disease," *Expert Systems with Applications*, vol. 41, no. 2, pp. 478–488, 2014. DOI: 10.1016/j.eswa.2013.07.073. 11, 36

[117] E. Sazonov, *Wearable sensors: fundamentals, implementation and applications*. Academic Press, an imprint of Elsevier, 2021. 6

[118] J. D. Schaafsma, Y. Balash, T. Gurevich, A. L. Bartels, J. M. Hausdorff, and N. Giladi, "Characterization of freezing of gait subtypes and the response of each to levodopa in parkinson's disease," *European Journal of Neurology*, vol. 10, no. 4, pp. 391–398, 2003. DOI: 10.1046/j.1468-1331.2003.00611.x. 5, 6

[119] Y. S. Shin and J.-J. Jeon, "Pseudo wigner–ville time-frequency distribution and its application to machinery condition monitoring," *Shock and Vibration*, vol. 1, no. 1, pp. 65–76, 1993. DOI: 10.1155/1993/372086. 84

[120] E. A. Shipton, "Movement disorders and neuromodulation," *Neurology Research International*, vol. 2012, pp. 1–8, 2012. DOI: 10.1155/2012/309431. 1

[121] L. Sigcha, N. Costa, I. Pavón, *et al.*, "Deep learning approaches for detecting freezing of gait in parkinson's disease patients through on-body acceleration sensors," *Sensors*, vol. 20, no. 7, p. 1895, Mar. 2020. DOI: 10.3390/s20071895. [Online]. Available: https://doi.org/10.3390/s20071895. 18

[122] A. L. Silva de Lima, L. J. Evers, T. Hahn, *et al.*, "Freezing of gait and fall detection in parkinson's disease using wearable sensors: A systematic review," *Journal of Neurology*, vol. 264, no. 8, pp. 1642–1654, 2017. DOI: 10.1007/s00415-017-8424-0. 5

[123] P. J. Silvia Mangia, "Magnetic resonance imaging (mri) in parkinson's disease," *Journal of Alzheimer's Disease &; Parkinsonism*, vol. 03, no. 03, 2013. DOI: 10.4172/2161-0460.s1-001. 2

[124] K. Simonyan and A. Zisserman, *Very deep convolutional networks for large-scale image recognition*, 2015. arXiv: 1409.1556 [cs.CV]. 30, 53

[125] P. Sollich and A. Krogh, "Learning with ensembles: How overfitting can be useful.," vol. 8, Jan. 1995, pp. 190–196. 8

[126] S. Soltaninejad, I. Cheng, and A. Basu, "Kin-FOG: Automatic simulated freezing of gait (FOG) assessment system for parkinson's disease," *Sensors*, vol. 19, no. 10, p. 2416, May 2019. DOI: 10.3390/s19102416. [Online]. Available: https://doi.org/10.3390/s19102416. 14, 16

[127] *Spm software - statistical parametric mapping*. [Online]. Available: https://www.fil.ion.ucl.ac.uk/spm/software/ (visited on 06/18/2021). 24

[128] *Statistics*. [Online]. Available: https://www.parkinson.org/Understanding-Parkinsons/Statistics. 2

[129] T. Steffen and M. Seney, "Test-retest reliability and minimal detectable change on balance and ambulation tests, the 36-item short-form health survey, and the unified parkinson disease rating scale in people with parkinsonism," *Physical Therapy*, vol. 88, no. 6, pp. 733–746, 2008. DOI: 10.2522/ptj.20070214. 106

[130] *Structural brain mapping group*. [Online]. Available: http://www.neuro.uni-jena.de/ (visited on 06/18/2021). 24

[131] S. Sveinbjornsdottir, "The clinical symptoms of parkinson's disease," *Journal of Neurochemistry*, vol. 139, pp. 318–324, 2016. DOI: 10.1111/jnc.13691. 3, 6

[132] C. M. Tanner and S. M. Goldman, "Epidemiology of parkinson's disease," *Neurologic clinics*, vol. 14, no. 2, p. 317, 1996. 2

[133] H. Teräväinen, L. Forgach, M. Hietanen, M. Schulzer, B. Schoenberg, and D. Calne, "The age of onset of parkinson's disease: Etiological implications," *Canadian journal of neurological sciences*, vol. 13, no. 4, pp. 317–319, 1986. 2

[134] "The parkinson progression marker initiative (ppmi) - experience with data and biospecimen access (p06.083)," *Neurology*, vol. 78, no. Meeting Abstracts 1, 2012. DOI: 10.1212/wnl.78.1_meetingabstracts.p06.083. [Online]. Available: https://www.ppmi-info.org/ (visited on 06/18/2021). 22

[135] E. E. Tripoliti, A. T. Tzallas, M. G. Tsipouras, *et al.*, "Automatic detection of freezing of gait events in patients with parkinson's disease," *Computer Methods and Programs in Biomedicine*, vol. 110, no. 1, pp. 12–26, Apr. 2013. DOI: 10.1016/j.cmpb.2012.10.016. [Online]. Available: https://doi.org/10.1016/j.cmpb.2012.10.016. 15

[136] T. Vos, C. Allen, M. Arora, *et al.*, "Global, regional, and national incidence, prevalence, and years lived with disability for 310 diseases and injuries, 1990–2015: A systematic analysis for the global burden of disease study 2015," *The Lancet*, vol. 388, no. 10053, pp. 1545–1602, 2016. DOI: 10.1016/s0140-6736(16)31678-6. 2

[137] J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, "Deep learning for sensor-based activity recognition: A survey," *Pattern Recognition Letters*, vol. 119, pp. 3–11, 2019. DOI: 10.1016/j.patrec.2018.02.010. 7, 8

[138] K. Wirdefeldt, H.-O. Adami, P. Cole, D. Trichopoulos, and J. Mandel, "Epidemiology and etiology of parkinson's disease: A review of the evidence," *European Journal of Epidemiology*, vol. 26, no. S1, pp. 1–58, 2011. DOI: 10.1007/s10654-011-9581-6. 2, 3

[139] "Wordnet: An electronic lexical database. christiane fellbaum," *The Library Quarterly*, vol. 69, no. 3, pp. 406–408, 1999. DOI: 10.1086/603115. 12, 13

[140] A. Worker, C. Blain, J. Jarosz, *et al.*, "Cortical thickness, surface area and volume measures in parkinson's disease, multiple system atrophy and progressive supranuclear palsy," *PLoS ONE*, vol. 9, no. 12, 2014. DOI: 10.1371/journal.pone.0114167. 21

[141] H. Zach, A. M. Janssen, A. H. Snijders, *et al.*, "Identifying freezing of gait in parkinson's disease during freezing provoking tasks using waist-mounted accelerometry," *Parkinsonism & Related Disorders*, vol. 21, no. 11, pp. 1362–1366, Nov. 2015. DOI: 10.1016/j.parkreldis.2015.09.051. [Online]. Available: https://doi.org/10.1016/j.parkreldis.2015.09.051. 15

[142] Y. Zhang, M. Brady, and S. Smith, "Segmentation of brain mr images through a hidden markov random field model and the expectation-maximization algorithm," *IEEE Transactions on Medical Imaging*, vol. 20, no. 1, pp. 45–57, 2001. DOI: 10.1109/42.906424. 25

[143] Z.-H. Zhou, J. Wu, and W. Tang, "Corrigendum to "ensembling neural networks: Many could be better than all" [artificial intelligence 137 (1–2) (2002) 239–263]," *Artificial Intelligence*, vol. 174, no. 18, p. 1570, 2010. DOI: 10.1016/j.artint.2010.10.001. 88