Application of Machine Learning in the Diagnosis of Parkinson's Disease

Cameron Harwood

PhD

University of York

Physics, Engineering and Technology

October 2023

Abstract

Parkinson's Disease (PD) is a progressive neurodegenerative disorder characterised by both motor impairment and non-motor symptoms, including cognitive impairment. PD presents significant challenges for reliable diagnosis and accurate symptom assessment. The current "gold standard" clinical assessments rely on visual judgement, introducing subjectivity. This thesis aims to mitigate these limitations by applying objective machine learning methodologies to two distinct types of movement data, simple hand motor tasks and neuropsychological graphmotor assessments, with the objective of modeling motor severity and a potential for a more granular approach for assessing cognitive impairment for people with PD.

For the hand motor tasks, end-to-end time series classification models were used to analyse positional data collected from 47 healthy controls and 148 PD patients. These models were applied for the diagnosis of PD and for the detection of clinically slight bradykinesia. After employing a 5-fold nested cross-validation strategy, the top-performing models achieved an accuracy rate of 84% for PD diagnosis and 82% for bradykinesia detection. These models provide an agile, objective, and rapid framework for hand kinematic assessments, negating the need for domain-specific knowledge. They have the potential to serve as essential tools for preliminary research in the field of kinematic evaluations.

For the drawing assessments, the structural components of the Benson Complex Figure were identified with a top accuracy rate of 96%, following the novel investigation of encoding pen-dynamics. This enables the extraction of cognitive features related to the organisational strategy employed by the subjects.

Collectively, these findings introduce promising new data-driven approaches for the modeling of PD diagnosis and cognitive states. Importantly, the research is designed with the aim of integrating these methodologies into routine clinical practice and aligning with current research interests, thus laying the groundwork for future domain-specific studies in PD assessment.

Contents

List of Figures

List of Tables

Acknowledgements

First and foremost, I express my deepest gratitude to Prof. Stephen Smith. His encouragement and support have been deeply appreciated while on this PhD journey. His mentorship has extended beyond academic guidance, providing me with invaluable opportunities for personal and professional growth. Additionally, I thank Dr. David Halliday for his invaluable contributions and feedback.

I will remain continually grateful to my friends, family, and my partner Millie for their enduring support.

Declaration

I declare that this thesis is a presentation of original work and I am the sole author. This work has not previously been presented for a degree or other qualification at this University or elsewhere. All sources are acknowledged as references.

Chapter 1

Introduction

1.1 Clinical Motivation

PD has emerged as a pressing healthcare issue, as evidenced by its rapidly growing prevalence and associated societal burden. According to the Global Burden of Disease study spanning 1990 to 2016 [57], neurological conditions have become the leading cause of disability worldwide. Within this spectrum of disorders, PD has shown the most significant increase in age-standardised prevalence, disability, and mortality rates. Dorsey et al. extend this observation by characterising the escalating trend in PD as resembling a pandemic. They identify multiple factors potentially responsible for this rise, including aging populations, increased life expectancy, declining rates of smoking, and industrialisation by-products. These factors, either singly or in a combinatory fashion, may underlie the growing number of individuals diagnosed with PD [51]. Given this, the underlying pathological causes of PD are still not fully understood [153].

What is known however, is that symptoms arise from the progressive loss of dopaminergic neurons in the substantia nigra of the midbrain, resulting in the depletion of dopamine in the basal ganglia [72]. This biochemical imbalance, of which is also seen in dopamine antagonist drugs [170], manifests as the cardinal motor symptoms, collectively referred to as Parkinsonisms. These include akinesia and bradykinsia (loss and slowness of movement, respectively), tremor, and rigidity. People with Parkinson's Disease (PwPD) may also experience additional motor deficits such as gait disturbance, impaired handwriting, grip force, and quieter speech [81]. In addition to motor symptoms, PD is also associated with various non-motor symptoms, including urinary dysfunction, fatigue, depression, sleep disorders, and cognitive impairments [108]. Mild Cognitive Impairment (MCI), the stage between the expected decline in cognition due to aging and dementia, is common even in de novo PD; present in 25% to 30 % of non-demented patients with PD. Additionally, the transition from MCI to dementia is common (nearly 80%)[23]. Current treatments do not offer a cure for PD or cognitive dysfunction; however, early interventions for motor symptoms and cognitive decline have been correlated with improved patient outcomes [121, 23]. Furthermore, with no definitive test for identifying PD or MCI, diagnosis relies on clinically administered assessments [101].

The Movement Disorder Society - Sponsored Revision of the Unified Parkinson's Disease Rating Scale (MDS-UPDRS) is the "gold standard" for comprehensive evaluation of both motor and non-motor symptoms associated with PD [61]. The outcomes derived from this assessment are pivotal in clinical trials focused on the exploration of new treatments [10, 26]. Among these outcomes, upper-limb tests such as finger tapping are used to elicit bradykinesia symptoms. However, it should be noted that these assessments, although comprehensive, have limitations. They are largely subjective in nature, relying on clinician observation, and yield only moderate interrater reliability [197]. It should be noted for an assessment of MCI, it is recommended that supplementary neuropsychological evaluations be conducted to obtain a comprehensive cognitive profile of the patient [102].

The Rey-Osterrieth Complex Figure Test (ROCF) [151] is a widely used neuropsychological graphmotor (drawing) assessment, discussed further in Chapter 5, which can be summarised as assessing the following cognitive domains:

- **Visuospatial Abilities** : The test requires the individual to draw a complex figure, thereby assessing spatial perception and organisation. Using the figure it has been shown that individuals with autism have an atypical pattern of visual processing [93].
- **Memory** : The test often includes a delayed recall phase, reproducing the figure from memory.
- **Executive Functioning** : The task involves elements of planning and organizational skills, as the individual needs to decide how to approach drawing the complex figure.
- **Attention** : Completing the figure requires sustained attention to detail and the ability to focus on the task at hand.
- **Motor Skills**: Though not a primary domain of assessment, the act of drawing the figure also involves fine motor skills.

This test has specific relevance for PD, especially when considered in light of the 'dual syndrome' hypothesis [85]. According to this hypothesis, two cognitive subtypes can be distinguished in Parkinson's Disease Mild Cogntive Impairment (PD-MCI). A frontostriatal subtype characterised by deficits in **attention** and/or **executive functions** and a posterior cortical subtype primarily exhibiting impairments in **visuospatial** skills, **memory**, and/or language. The latter subtype has been associated with a higher risk of developing dementia. However, the standard scoring methods applied do not quantify the process or how the task was completed which could reveal deficits in domains relevant to each subtype.

The United Kingdom faces a notable shortage of consultant neurologists, which leads to extended waiting periods for PwPD to receive proper assessments [128]. The limitations of current diagnostic methods further compound the issue; clinician-based diagnoses may initially exhibit accuracy rates as low as 75-80% [154], although these rates typically improve with subsequent follow-up appointments [107]. This scarcity of specialised medical professionals, coupled with the inherent inadequacies of traditional scoring systems and diagnostic challenges, underscores the urgent need for a more efficient and reliable assessment methodology. To address this gap, the thesis applies digital sensors and data-driven models as a means to augment the objectivity and utility of clinical assessments.

1.2 Research Interest and Objectives

Recent advances in computer vision technologies have been increasingly recognised within the medical community for their capabilities in quantifying hand kinematics through video data [172]. Traditional machine learning approaches designed to objectively evaluate motor severity and for disease diagnosis, require domain-specific expertise to derive features of clinical relevance. Typically, high-dimensional hand kinematic data is often reduced to a one-dimensional signal. For example, the amplitude between the finger and the thumb is commonly used. However, this reduction inherently narrows the search space, potentially missing valuable data. It is noteworthy that only recently has the field begun to explore the extraction of features from 3D positional data in the context of the finger-tapping test [99]. Other motor tasks remain to be explored, highlighting that there is a need to establish which tasks are the most clinically relevant before domain-specific analysis occurs.

Deep learning, is an approach from which feature extraction is automatically performed, and has demonstrated comparable results to domain specific approaches [207]. Recent algorithmic advances in the space of multivariate time series have produced end-to-end algorithms that do not rely on gradient based optimisation, and are very fast to train even on a CPU [45].

With the ease of data collection, and the ease of training end-to-end classification algorithms with minimal pre-processing, makes the possibility of applying this technique to a broader range of hand kinematic studies, of clinical interest.

Thus, as addressed in Chapter 4, the objective is to:

Evaluate the performance of end-to-end multivariate time-series classification algorithms for the prediction of PD, and motor severity within standard upper limb assessments

Digital graphics tablets have historically been employed for the objective assessment of PD, as expectedly motor symptoms manifest in graphmotor tasks [203, 31]. However, the typical tests applied do not directly relate to cognitive domains. For tasks like the ROCF, manual approaches to recording organisation and strategy have long existed, although similarly to upper limb assessment, the granularity in which organisation and planning can be objectively recorded is limited. Recent investigations into the semantic evaluation of these tasks are promising [139], although they are limited in their practical applicability due to the requisite manual segementation of strokes.

Thus, as addressed in Chapter 5 the objective is to:

Develop an approach for the automatic segmentation of neuropsychological drawings

Subsequent to this primary focus, a supplementary line of inquiry emerged, investigating feature representations that enable the inclusion of pen-dynamics within these segmented drawings.

The overarching hypothesis of this work is then:

"Machine learning methodologies can serve as effective tools in improving diagnostic utility of standard clinical assessments in Parkinson's Disease"

This aligns with the broader goal of developing objective, efficient, and cost-effective diagnostic tools that can improve the diagnostic utility of routine clinical assessments. The research question will be revisited in the conclusion, leveraging the empirical results.

1.3 Structure of Thesis

Chapter 2 provides a background to PD and clinical assessments. Chapter 3 details the evaluation methods utilised during the later chapters. Chapter 4 provides the first evaluation of recent timeseries classification methods for the diagnosis of PD and bradykinesia, using positional data collected from three distinct upper limb assessments. Chapter 5 details the proposed approach for digital drawing segmentation, with the investigation on whether encoding pen dynamics will assist in segmentation. Finally Chapter 6 concludes the thesis, reviewing the rationale and work conducted, the key findings and outlining future work.

Chapter 2

Parkinson's Disease: Clinical Presentation and Assessment

The objective of this chapter is to provide the reader with an overview of PD, including its fundamental characteristics and relevant assessment methods. Firstly, the signs and symptoms of PD, treatments available, along with challenges in its diagnosis, particularly in relation to other similar conditions, will be explored. Secondly the "gold standard" assessment tools used in the diagnosis and assessment of PD, emphasising their clinical relevance and scoring systems employed will be outlined. This section will offer insights into the most reliable and widely accepted methods for evaluating the disease.

2.1 Clinical Presentation of Parkinson's Disease

References to symptoms resembling those of PD can be found in ancient texts, substantiating the disease's longstanding impact on human health [206, 132]. It was James Parkinson's seminal work "An Essay on the Shaking Palsy" published in the early 19th century which laid the foundation for the contemporary understanding of the disease named in his honour. Parkinson defined the condition as:

> *"Involuntary tremulous motion, with lessened muscular power, in parts not in action and even when supported; with a propensity to bend the trunk forwards, and to pass from a walking to a running pace: the senses and intellects being uninjured" [135]*

Main Symptoms (often referred to as Parkinsonisms)

- * Tremor - Uncontrollable shaking or trembling; usually begins in the hand or arm when relaxed and resting
- * Rigidity (muscle stiffness) - tension in the muscles, impeding mobility and can result in painful muscle cramps (dystonia)
- Bradykinesia (slowness of movement) - physical movements are distinctively slowed, resulting in a shuffling gait
- * Postural instability - balance problems, increasing the likelihood of falls and injuries

Physical Symptoms

- * Constipation
- * Excessive production of saliva
- * Dysphagia (problems swallowing)
- * Insomnia
- Anosmia (loss of sense of smell) - may occur in the prodomal phase, many years before the onset of symptoms
- Nerve pain
- Urinary incontinence
- Sexual dysfunction
- Dizziness, blurred vision or fainting - caused by a sudden drop in blood pressure
- Hyperhidrosis (excessive sweating)

Cognitive and Psychiatric Symptoms

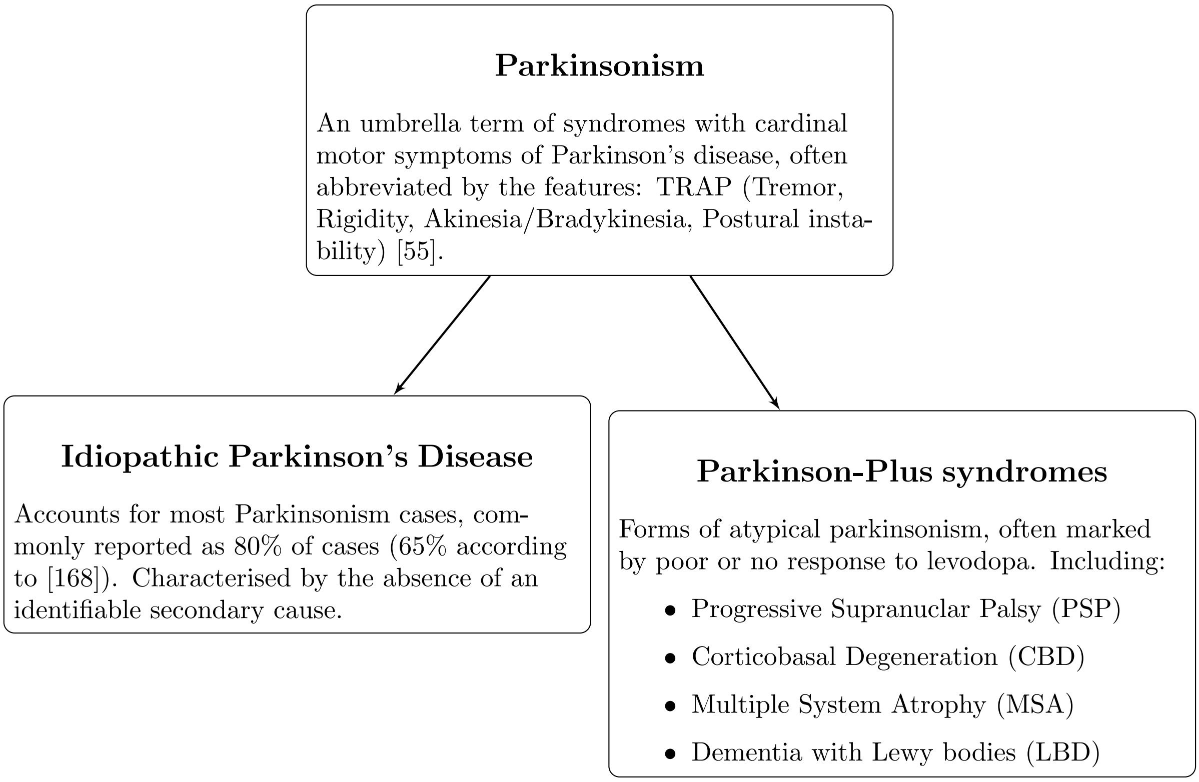
- Depression and anxiety
- Mild cognitive impairment - slight memory problems and problems with activities that require planning and organisation
- Dementia - a group of symptoms, including more severe memory problems, personality changes, seeing things that are not there (visual hallucinations) and believing things that are not true (delusions)

Table 2.1: Symptoms of Parkinson's Disease adapted from NHS guidelines [126]. A * denotes symptoms originally identified by James Parkinson [135].

Parkinson's observations encompass several primary symptoms that have remained central to diagnosis today, as detailed in Table 2.1. Much of the historical and current diagnostic framework has focused on the motor manifestations of the disease. One such refinement, from Jean-Martin Charcot in the late 19th century was the identification of Bradykinesia [30]. This symptom, defined as the "slowness of initiation of voluntary movement with progressive reduction in speed and amplitude of repetitive actions", is particularly notable as it must be present (in addition to one other primary Parkinsonism symptoms) to meet the UK Parkinson's Disease Society Brain Bank diagnostic criteria [35]. These criteria aim to maximise diagnostic accuracy during a patient's lifetime and are particularly useful for standardising the classification of subjects in research settings. The overarching aim of these guidelines is to approximate, through clinical means, the conclusive neuropathological diagnosis that can only be confirmed via post-mortem tissue examination [74]. In achieving this, the criteria affords clinicians the opportunity to initiate timely and precise interventions. However, early diagnosis of PD remains challenging, with misdiagnosis rates reported at 25% [154], largely owing to the fact that a number of other conditions share the cardinal symptoms, illustrated by Figure 2.1.

Neuropathological hallmarks of PD include structural changes such as the loss of dopaminergic cells within the substantia nigra, and the buildup of alpha-synuclein proteins, forming structures known as Lewy bodies and Lewy neurites inside nerve cells. By the time PD symptoms manifest clinically, a large number of these dopamine-producing cells have already been lost; Toss et al. [159] report a 50% loss. The main motor symptoms are attributed to the death of nerve cells in the substantia nigra, a section of the midbrain critical for the production of the neurotransmitter dopamine. The reduction in overall dopamine levels negatively impacts the communication ability of the basal ganglia, "a group of sub-cortical nuclei responsible primarily for motor control, as well as other roles such as motor learning, executive functions and behaviours and emotions" [95]. The underlying pathology for this cell loss remains unclear, but it is associated with the aforementioned buildup of alpha-synuclein and formation of Lewy bodies and neurites.

Ultimately, PD is neurodegenerative, as neurons gradually die or diminish in their function, new symptoms emerge and existing ones often intensify. For instance, initial motor symptoms generally occur asymmetrically, affecting one side of the body more than the other. As it progresses, the motor symptoms may become bilateral, but one side of the body often remains more affected than the other [50]. Introduced in 1967, the Hoehn and Yahr scale [69], was the first objective measure for assessing the progression of the disease. This scale consists of five stages, ranging from mild to severe, based on the degree of functional impairment and the presence of bilateral involvement. The modified Hoehn and Yahr stage is detailed in Table



Parkinsonism

An umbrella term of syndromes with cardinal motor symptoms of Parkinson's disease, often abbreviated by the features: TRAP (Tremor, Rigidity, Akinesia/Bradykinesia, Postural instability) [55].

Idiopathic Parkinson's Disease

Accounts for most Parkinsonism cases, commonly reported as 80% of cases (65% according to [168]). Characterised by the absence of an identifiable secondary cause.

Parkinson-Plus syndromes

Forms of atypical parkinsonism, often marked by poor or no response to levodopa. Including:

- Progressive Supranuclar Palsy (PSP)
- Corticobasal Degeneration (CBD)
- Multiple System Atrophy (MSA)
- Dementia with Lewy bodies (LBD)

Figure 2.1: Overview of Parkinsonism and related disorders

2.2.

Although the Hoehn and Yahr scale provides a clinical framework for assessing disease progression, the underlying pathological mechanisms remain less clear. The Braak hypothesis has gained attention for its attempt to address this gap [14]. Braak hypothesised that an unknown pathogen in the gut could be the cause of PD. The system divides the disease into six stages based on the distribution of Lewy bodies. Stages 1 - 2 focus on the olfactory structures and the vagus nerve, reflecting the intitial impact on smell. Stages 3-4 involve the spread to the midbrain and basal ganglia, affecting movement, while stages 5-6 reach the cortical regions impacting cognitive functions. Higher Braak stages were associated with cognitive decline and dementia [37].

Despite James Parkinson's insights into non-motor symptoms, much of the historical and current diagnostic framework has centered on the motor manifestations of the disease. In a recent survey, 48% of people with PD reported that non-motor symptoms presented a greater challenge for quality of life than motor symptoms [68].

Parkinson's Disease Dementia (PD-D) is a cognitive impairment that can develop in individuals diagnosed with PD. It is characterised by a decline in cognitive functions that is more pronounced than what might be expected from normal aging or the typical progression of PD itself. PD-D often manifests as deficits in attention, executive function, memory, and visuospatial abilities. These impairments can affect daily living activities and reduce the quality of life. In a systematic review, the point prevalence of dementia was found to be 24-31% [1].

PD-D is closely related to Lewy Body Dementia (LBD). The distinction between the two is that cognitive issues are more prevalent earlier on in LBD. However, it should be noted that the association between cortical Lewy body pathology and dementia is not consistently observed in all cases.

MCI is an increasingly recognised non-motor symptom of Parkinson's disease. Similar to PD-D, PD-MCI affects the ability to perform the activities of daily living but is less severe. People with PD-MCI are at a greater risk of transitioning to dementia. Alterations to lifestyle may play important roles in preventing of slowing down this conversion; however, no current treatments slow down or halt the progression of PD.

Stage	**Description**
1.0	Unilateral involvement only.
1.5	Unilateral and axial involvement.
2.0	Bilateral involvement without impairment of balance.
2.5	Mild bilateral involvement with recovery on retropulsion (pull) test.
3.0	Mild to moderate bilateral involvement, some postural instability but physically independent.
4.0	Severe disability, still able to walk and to stand unassisted.
5.0	Wheelchair bound or bedridden unless aided.

Table 2.2: Modified Hoehn and Yahr Stages [59]

It was through the work of Nobel laureate Arvid Carlsson, whereby the vital link between dopamine and Parkinson's disease was first discovered, and from which the most effective treatment for Parkinson's disease is derived [25]. Carlsson's experiments in the late 1950s demonstrated that dopamine is a neurotransmitter in the brain and that it plays a crucial role in controlling movement. He showed that animals deprived of dopamine became immobile, but their movement could be restored with the amino-acid Levodopa (L-DOPA), a precursor to dopamine. When administered it is able to cross the blood-brain barrier and is then converted to dopamine in the brain. The restoration of dopamine significantly alleviates many of Parkinson's diseases motor symptoms. Today, oral L-DOPA is now the "gold standard" treatment, dramatically improving many patients quality of life. However it is not a cure, and can also lead to similar motor complications referred to as Levodopa-induced Dyskinesia (LID), that also worsens time.

An alternative treatment is an invasive surgical procedure called Deep Brain Stimulation (DBS) [36]. The procedure involves the insertion of two probes into the brain. They are then attached by wires to an impulse generator under the skin in the chest. The impulse generator emits signals to the probes which helps override the mechanical effects of PD. The reason why this overrides the tremors is not yet fully understood. Similar to LID, stimulators and medications should be optimised for the patient. People with underlying conditions who do not respond well to L-DOPA or who are simply too old, are not offered this treatment. An additional exclusion criteria is those with significant cognitive impairment [115]. Tritation for medication and for DBS can prove extremely challenging and time consuming.

2.2 Clinical Assessments

Concurrent with the advancements in treatments for PD, clinical assessments employed to validate their efficacy have also evolved. The advent of oral L-DOPA necessitated a robust measurement system; the Hoehn and Yahr scale fulfilled this role, providing a quantifiable means to assess the treatment's impact. In a similar vein, the Unified Parkinson's Disease Rating Scale (UPDRS) served as a critical instrument for the validation of DBS as a therapeutic intervention. These assessment tools not only validate the efficacy but also offer a standard against which the safety and patient outcomes can be evaluated. The subsequent sections delineate the most widely adopted assessment instruments in PD.

2.2.1 Motor

The MDS-UPDRS [61] serves as the current "gold standard" for assessing the severity and progression of Parkinson's disease. Established in 1985, the Movement Disorder Society (MDS) is a professional organisation comprised of clinicians, scientists and healthcare professionals specialising in movement disorders. The society plays a pivotal role in establishing clinical guidelines, integrating contemporary research and expert consensus. The MDS-UPDRS is well validated as a whole [144] and comprehensive, comprising four sections:

1. **Non-Motor Experiences of Daily Living** (13 questions) - This section consists of a combination of self reported and clinically derived measures that evaluate mood, cognition, behaviour and sleep quality. It aims to capture the non-motor symptoms often overlooked but significantly impacting the quality of life in PwPD.

2. **Motor Experiences of Daily Living** (13 questions) - Self reported measures of pertaining to complications with motor tasks in daily living such as dressing, eating and walking.

3. **Motor Examination** (33 scores based on 18 questions with several right, left or other

body distribution scores) - Clinically scored measures of motor function, including standardised assessments for tremor, rigidity and bradykinesia.

4. **Motor Complications** (6 questions) - Clinically scored measures of complications arising from dopaminergic therapy, such as motor fluctuations of dyskinesia.

In total, the MDS-UPDRS includes 65 scored measures, each rated on a scale between 0 (normal) to 4 (severe). Part III carries the most significance in the scoring system, key elements include:

- **Rigidity** (3.3) - Assessed by manually moving the patient's limbs and neck. The clinician evaluates resistance to passive movement and may also assess cogwheel rigidity (ratchet like movement).

- **Bradykinesia** (3.4 - 3.8) - This involves tests like finger tapping, hand movements, and rapid alternating movements. The clinician scores the patient based on speed, amplitude, rhythm and hesistations during the task.

- **Postural and Gait** (3.9 - 3.13) - This includes evaluations on the patient's posture, stability and ability to walk.

- **Tremor Assessment** (3.15 - 3.18) - Evaluates postural (against gravity), resting and kinetic (performed with an action) tremors.

Significantly (in relation to PD diagnosis), items pertaining to bradykinesia demonstrate the lowest reliability [67]. This is due in part to the inherent subjectivity in visual assessment, as well as the requirement to simultaneously aggregate several movement attributes (speed, amplitude, rhythm, hesitations) into a single score. This is most evident when the severity is slight or mild [60]. Further insights into the reliability of finger-tapping tests are provided by the study conducted by Stefan et. al. [197]. In this twenty one movement disorder neurologists were presented with an eleven second finger tapping video, and asked to rate the severity, critically without any cues from additional examination history. As noted in their conclusion, "even experts show considerable disagreement about the level of bradykinesia on finger tapping, and frequently see bradykinesia in the hands of those without neurological disease" [197].

2.2.2 Cognitive

The evaluation of cognitive symptoms, particularly milder manifestations, in PD remains inadequately addressed by the single cognitive item within the MDS-UPDRS. To provide a more nuanced understanding, the MDS has introduced specific guidelines for diagnosing PD-MCI. The guidelines are stratified into two levels of assessment that vary in comprehensiveness. Level I Assessments, such as the Montreal Cognitive Assessment (MoCA) [123], offer a rapid global evaluation of cognitive function. In contrast, Level II Assessments provide a more granular exploration of cognitive domains, including attention, executive function, language, and visuospatial abilities.

Among the tools employed for these assessments, constructional tasks, such as the reproduction of a 3D cube or interlocking pentagons, are widely favored. These tasks serve the dual purpose of evaluating motor coordination as well as cognitive function, particularly executive and visuospatial abilities. Simple tasks like the interlocking pentagons have demonstrated utility in identifying cognitive decline in PD [84].

As detailed in Table 2.3, various assessments utilise multiple scoring schemes, and none account for the copy strategy employed by the patient.

Test/Battery	Graphomotor Assessment	Relevant Metrics
GPCOG [18]	Clock Drawing Test	Numbers (0-1), Hands (0-1)
Mini-cog [13]	Clock Drawing Test	Pass/Fail
MMSE [92]	Interlocking Pentagons	Pass/Fail
ACE-III [11]	Various Tasks	Infinity Diagram (0-1), Wire Cube (0-2), Clock (0-5)
MoCA [123]	Various Tasks	Pentagons (0-1), Cube (0-1), Clock (0-3)
NACC UDS [116]	Benson Figure Copy and Recall	Accuracy and placement (0-17)

Table 2.3: Summary of graphmotor tasks within standard cognitive assessments

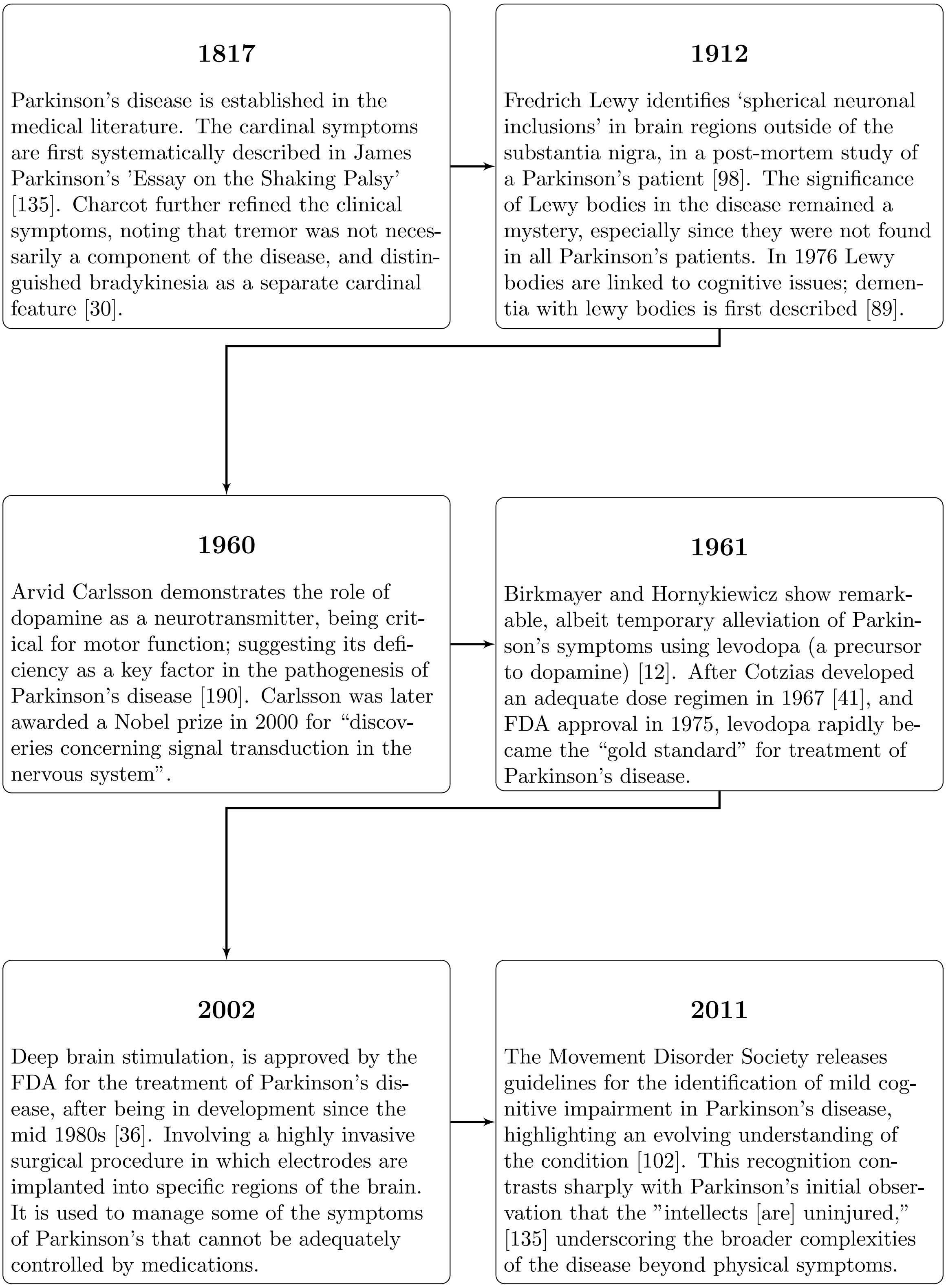
2.3 Current Application of Digital Sensors

In alignment with the NHS's overarching digital initiatives [4], the growing utilisation of digital sensors, capable of capturing a wide array of human movement, signifies a transformative shift in disease assessment methodologies. The National Institute for Health and Care Excellence (NICE) has already endorsed the use of five specific inertial sensors for home-based monitoring of PD [127].

These sensors are aimed with objective to monitoring bradykinesia, and LID in the home. Several of these devices employ machine learning algorithms to analyse extracted movement features [155, 7], estimating severity of dyskinesia and bradykinesia for medication titration purposes. However, it is crucial to note that the effectiveness of these technologies is contingent upon a prior clinical diagnosis.

2.4 Summary

- A timeline of key advancements in the progression of PD is presented in Figure 2.2.
- Current treatments cannot slow or stop the progression of the disease, but they can ease symptoms, and help people continue a good quality of life. As a result early diagnosis and accurate diagnosis is a crucial first step to successful monitoring of the disease in addition to providing appropriate treatment.
- Bradykinesia, the key symptom, it is assessed in the part III of the MDS-UPDRS. Scoring for these tasks is subjective.
- Visuoconstruction assessments play a role in the many global cognitive assessments. The scoring for each assessment is usually crude, and does not reflect all aspects of drawing.



1817

Parkinson's disease is established in the medical literature. The cardinal symptoms are first systematically described in James Parkinson's 'Essay on the Shaking Palsy' [135]. Charcot further refined the clinical symptoms, noting that tremor was not necessarily a component of the disease, and distinguished bradykinesia as a separate cardinal feature [30].

1912

Fredrich Lewy identifies 'spherical neuronal inclusions' in brain regions outside of the substantia nigra, in a post-mortem study of a Parkinson's patient [98]. The significance of Lewy bodies in the disease remained a mystery, especially since they were not found in all Parkinson's patients. In 1976 Lewy bodies are linked to cognitive issues; dementia with lewy bodies is first described [89].

1960

Arvid Carlsson demonstrates the role of dopamine as a neurotransmitter, being critical for motor function; suggesting its deficiency as a key factor in the pathogenesis of Parkinson's disease [190]. Carlsson was later awarded a Nobel prize in 2000 for "discoveries concerning signal transduction in the nervous system".

1961

Birkmayer and Hornykiewicz show remarkable, albeit temporary alleviation of Parkinson's symptoms using levodopa (a precursor to dopamine) [12]. After Cotzias developed an adequate dose regimen in 1967 [41], and FDA approval in 1975, levodopa rapidly became the "gold standard" for treatment of Parkinson's disease.

2002

Deep brain stimulation, is approved by the FDA for the treatment of Parkinson's disease, after being in development since the mid 1980s [36]. Involving a highly invasive surgical procedure in which electrodes are implanted into specific regions of the brain. It is used to manage some of the symptoms of Parkinson's that cannot be adequately controlled by medications.

2011

The Movement Disorder Society releases guidelines for the identification of mild cognitive impairment in Parkinson's disease, highlighting an evolving understanding of the condition [102]. This recognition contrasts sharply with Parkinson's initial observation that the "intellects [are] uninjured," [135] underscoring the broader complexities of the disease beyond physical symptoms.

Figure 2.2: Timeline of key advancements in Parkinson's Disease research and treatment.

Chapter 3

Supervised Machine Learning

3.1 Introduction

Machine learning constitutes the ability for algorithms to determine discriminative patterns from data. It is comprised of three broad categories: unsupervised learning, reinforcement learning, and supervised learning. Supervised learning is differentiated by the use of labels. The process of supervised learning involves training a model to make predictions by providing it with a set of features and their corresponding target labels. As depicted in Figure 3.1, an input is provided to a trained classifier, which then generates a prediction. The efficacy of these models lies in their ability to generalise from the training data to unseen instances, thereby making accurate predictions on new data.

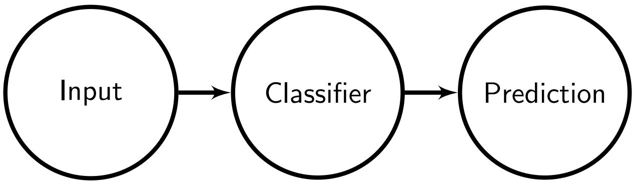


Figure 3.1: From input to prediction via a classifier.

There are two general tasks, regression and classification. In the case of regression tasks, the model attempts to predict continuous values, such as forecasting temperature or body height. In contrast, classification tasks, which are the focus of this work, involve predicting discrete labels, such as whether an email is 'spam' or not spam'.

Supervised learning algorithms have been successfully deployed in the field of healthcare for disease detection and diagnosis. In the context of Parkinson's disease, we can treat the diagnosis task as a binary classification task, where the two labels are 'Parkinson's' and 'healthy'.

As illustrated in Figure 3.2, most supervised learning algorithms employ a training loop to optimise internal parameters. The classifier generates predictions based on the input and current internal parameters; the predictions generated by the estimator are compared to the actual values to calculate a loss or error. The parameters of the estimator are then updated in a way that minimally reduces the loss. This iterative process continues until the model's predictions are as close as possible to the actual targets, or until further training no longer improves the model's performance.

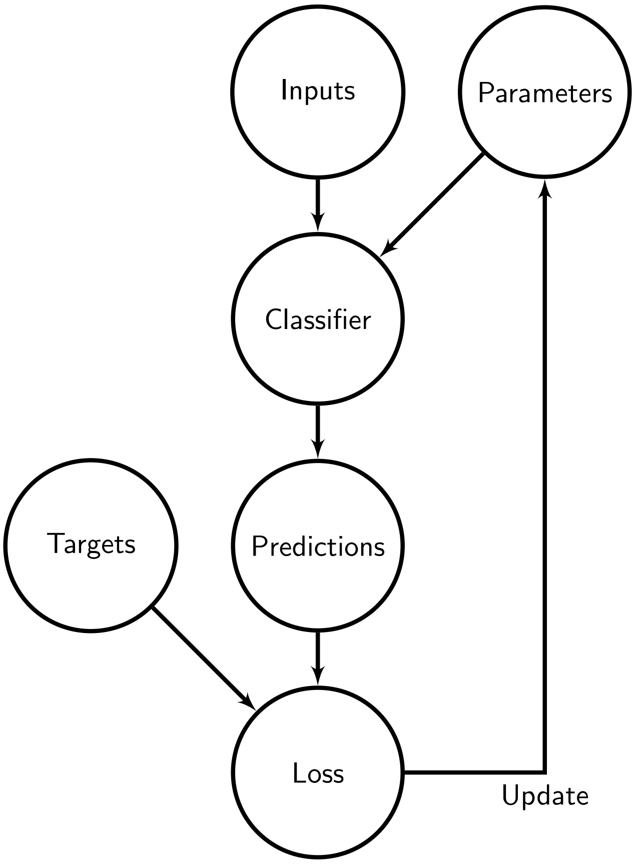
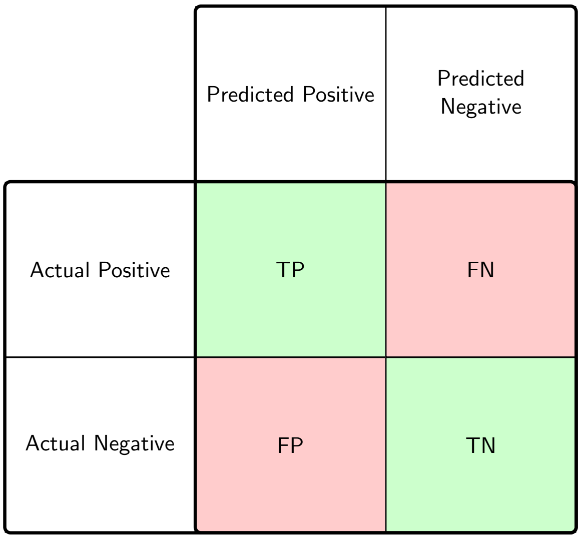


Figure 3.2: Training loop for machine learning algorithm.

3.2 Evaluation of Classifiers

This section will detail standard metrics used to assess machine learning classifiers in binary classification tasks as well as multi-class tasks, the latter is evaluated by breaking them down into a series of binary tasks using either the One-vs-One (OVO) or One-vs-Rest (OVR) strategy, followed by aggregation.

3.2.1 The Confusion Matrix, its Measures and Derived Metrics



	Predicted Positive	Predicted Negative
Actual Positive	TP	FN
Actual Negative	FP	TN

Figure 3.3: The Confusion Matrix, visualising four measures. The number of True Positives (TP) , False Negatives (FN), False Positives (FP), True Negatives (TN)

The confusion matrix (Fig 3.3), visualises the output from a binary classifier. From which one of the two classes is chosen to be the positive class. In a medical context, this generally refers to the presence of a condition, disease or attribute that the classifier aims to detect or predict. The positive class should be made explicit to ensure correct interpretation of results derived from the confusion matrix. The four possible measures are the following:

- **True Positives (TP)**: The number of positive cases predicted correctly.
- **True Negatives (TN)**: The number of negative cases predicted correctly.
- **False Positives (FP)**: The number of positive cases incorrectly predicted as positive. A False Positive is also known as a Type I error.
- **False Negatives (FN)**: The number of negative cases incorrectly predicted as negative. A False Negative is also known as a Type II error, and is more important medical diagnostic purposes, as failing to identify a disease may delay treatment.

The following metrics are derived from the confusion matrix measures, higher scores indicate better performance.

3.2.1.1 Accuracy

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \tag{3.1}$$

The ratio of correctly identified cases over the total number of cases, generally reported as a percentage. While being the most easily interpretable metric, it has two major shortcomings that should be considered. First and foremost, is the accuracy paradox [188], whereby in a imbalanced dataset, high accuracy can be acheived by simply predicting the majority class for all instances. Second, all correct predictions are considered equally, irrespective of class, and does not distinguish between false positives and false negatives.

3.2.1.2 Precision, Sensitivity (Recall) and Specificity

$$Precision = \frac{TP}{TP + FP} \tag{3.2}$$

Also known as the positive predictive value, precision measures the proportion of actual positive instances that are correctly identified by the model. Important in contexts where minimising false positive errors is crucial.

$$Sensitivity = \frac{TP}{TP + FN} \tag{3.3}$$

Also known as the **true positive tate (TPR)** or **recall**, sensitivity quantifies the ability of a classifier to correctly identify positive cases from all positive cases. High sensitivity is crucial in healthcare diagnostics to correctly identify most individuals with a condition, albeit with at the risk of a higher **false positive rate** (abbreviated as **FPR**, is the proportion of negative instances that are incorrectly identified as positive).

$$Specificity = \frac{TN}{TN + FP} \tag{3.4}$$

Also known as the true negative rate, specificity measures the proportion of actual negative instances that are correctly identified by the model. The FPR, is given as 1 - Specificity.

3.2.1.3 F1-Score

$$F1\text{-}score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{3.5}$$

Often used in lieu of accuracy, the F1-score is defined as the harmonic mean of precision and recall. It provides a balance between both measures, which are weighted equally. The F1-score ranges between 0 and 1, where a score of 1 indicates perfect precision and recall, and 0 indicates the worst possible score. The F1-score is independent to the number of samples correctly classified as negative, and thus with precision and recall, varies based on the positive class.

3.2.1.4 Matthew's Correlation Coefficient

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FN).(TP + FN).(TN + FP).(TN + FN)}} \tag{3.6}$$

The Matthew's Correlation Coefficient (MCC) is recommended over the F1-Score for binary classification evaluation, since it takes into account all four measures of the confusion matrix [34, 33]. The MCC treats the actual class and predicted class as two variables and computes their correlation coefficient, making it a robust metric for imbalanced datasets. The MCC returns a value between -1 and 1. A value of 1 represents a perfect prediction, 0 represents a random prediction, and -1 indicates total disagreement between predicted values and actual values.

3.2.1.5 The Receiver Operating Characteristic (ROC) Area Under the Curve (AUC)

In binary classification, classifiers often output the predicted probability for the positive class. As a standard, a threshold of 0.5 is used, but it can be tuned to improve sensitivity or specificity but not both (for less than perfect models). The Receiver Operating Characteristic (ROC) is a graphical representation of varying the threshold as presented by Figure 3.4.

The Area Under the Curve (AUC) quantifies the classifiers overall ability to discriminate between positive and negative classes.

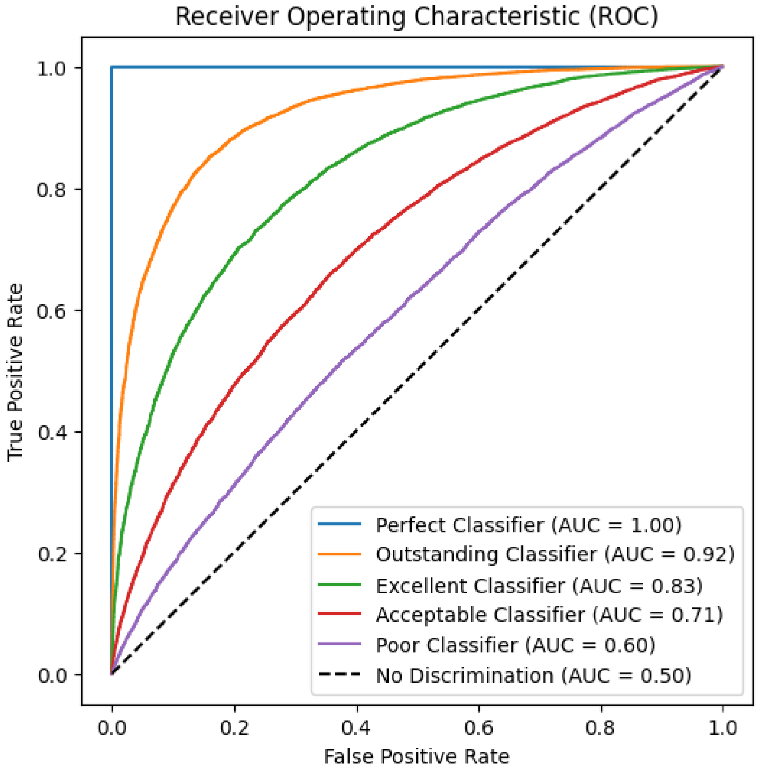


Figure 3.4: ROC curves for four simulated classifiers demonstrating different levels of discrimination ability based on their AUC scores (following rule of thumb from Hosmer and Lemeshow [97]).

3.2.2 Metrics for the Semantic Segmentation of Images

The goal of semantic segmentation is to understand the contents of images. Specifically pixel-level classification of images, where each pixel is assigned to a specific class. Recently, deep learning methods, in particular Convolutional Neural Networks (CNNs), have achieved significant improvements in this field [103].

The dataset of line drawings, such as that presented in Figure 3.5 in this thesis have am additional consideration when calculating performance metrics. Given that the background class dominates the image, as discussed in Section 3.2.1.1, this will give a misleading estimate of accuracy. Thus performance evaluations in this work exclude the background class.

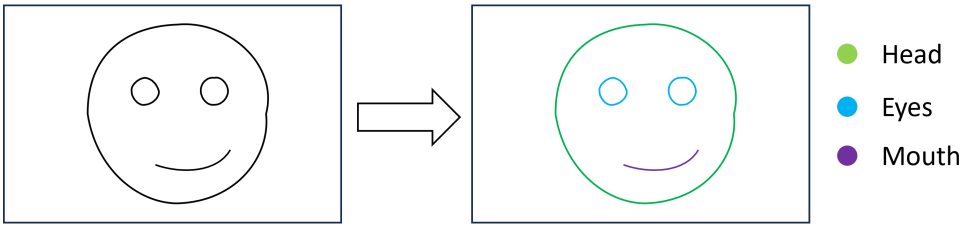


Figure 3.5: Semantic segmentation example, whereby each component of the image has been labelled.

3.2.2.1 Jaccard Similarity Coefficient

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{3.7}$$

Also known as the Intersection over Union (IoU) score, the Jaccard similarity coefficient quantifies the extent of overlap between the predicted segmentation and the ground truth segmentation.

3.2.3 Generalisability

For any classification task, we assume that there is some hidden process that is generating the data. Thus, the goal of a machine learning classifier is to predict well on sampled set of data from this hidden distribution. Real world data often includes some form of noise, as illustrated in Figure 3.6.

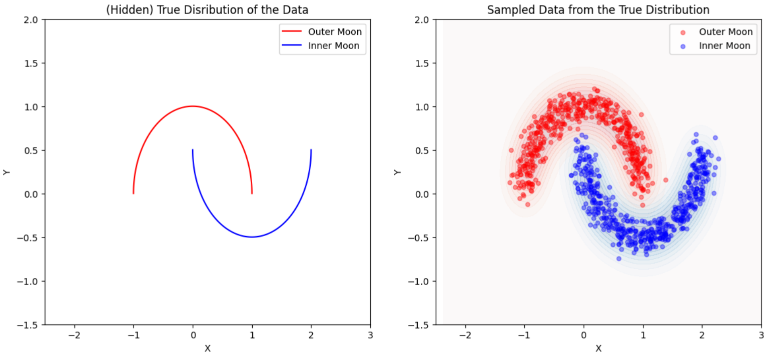


Figure 3.6: The left panel illustrates the true underlying distribution of the data (make moons synthetic dataset from sci-kit learn [136]). In right panel shows 1000 data points that sampled from this distribution, with the addition of Gaussian noise (standard deviation = 0.1) to simulate irreducible error. Kernel Density Estimation (KDE) with a Gaussian kernel and bandwith of 0.2 is employed to estimate the density of each class (on a larger sampled set of 10,000 data points), the contours of which represent the probability density of each class.

The generalisability of models is viewed through the lens of bias and variance. Bias refers to the systemic deviations of the model's predictions from the actual values, often a result of oversimplified assumptions and leading to underfitting. Conversely, variance reflects the model's sensitivity to fluctuations in the training data, causing overfitting when high. The tension between these two aspects is captured by the bias-variance trade-off, which requires a balancing the model's complexity [64].

Cross-validation serves as an effective technique for assessing a model's generalisation performance, thereby mitigating the risks associated with overfitting. In k-fold cross-validation,

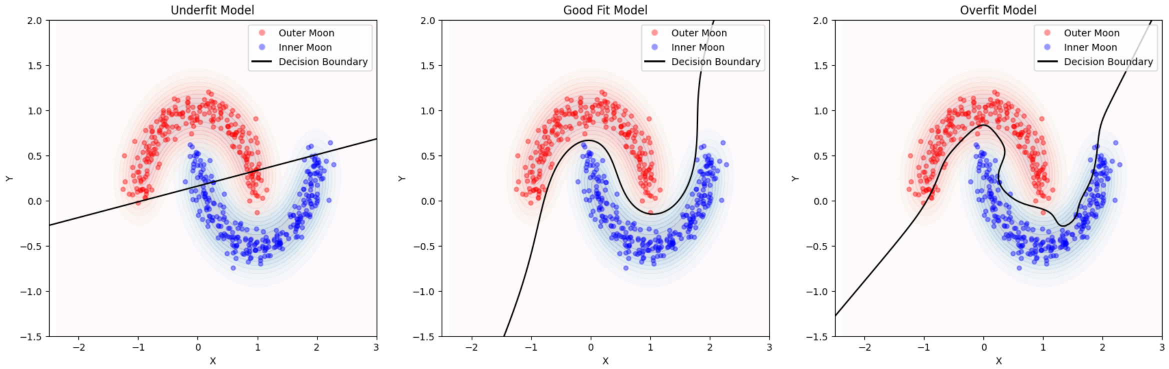


Figure 3.7: This figure shows the decision boundaries (black line) for three classifiers with varying hyperparameters that introduce increasing complexity. From 1,000 data points 500 were used to train the classifier, and the remaining 500 unseen data samples are plotted to visualise the model's performance on new data. The classifier in the left panel suffers from high bias (overly simplistic assumptions), and is failing to capture the underlying structure of the data. The classifier in the middle panel presents a well balanced model and offers the most generalisable performance. Whereas the classifier on the right is showing higher variance, capturing noise in the dataset.

data is partitioned into k subsets. The model is trained k times, using $k-1$ subsets for training and one for validation, with the average performance metric indicating generalisation capability. Nested cross-validation further refines this by including an inner loop for hyperparameter tuning and an outer loop for model assessment. The inner loop performs k-fold cross-validation for each hyperparameter set, and the best-performing set is chosen. This process is independent for each outer fold, ensuring unbiased evaluation. The outer loop then averages the performance measures, giving an unbiased estimate of model generalisation. Examples of cross-validation and nested cross-validation is given in Figures 3.8 and 3.9 respectively.

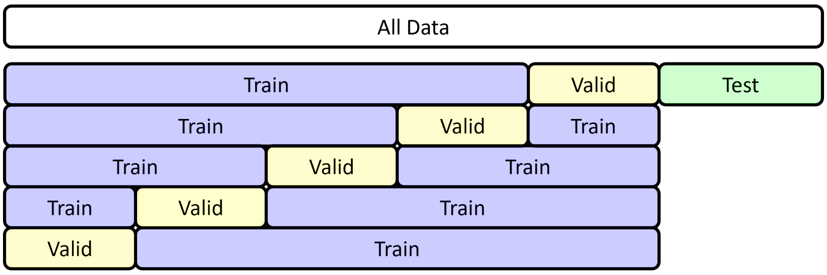


Figure 3.8: 5-fold cross validation example

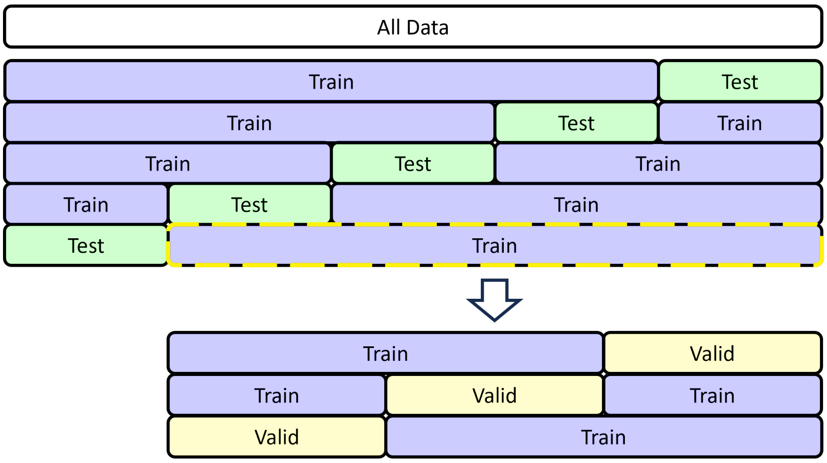


Figure 3.9: Nested cross validation with 5 outer folds and 3 inner folds

3.3 Traditional Machine Learning Techniques

3.3.1 Linear Regression

Linear regression is one of the most fundamental and widely used statistical methods in data analysis and machine learning. Dating back to the early 19th century, it has stood the test of time due to its simplicity, interpretability, and effectiveness in modeling relationships between variables [179].

At its core, linear regression aims to model the relationship between a dependent variable and one or more independent variables by fitting a linear equation to observed data. The basic form of this equation is:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n \tag{3.8}$$

Where y is the dependent variable, $x_1, x_2, ..., x_n$ are independent variables (features), $\beta_1, ..., \beta_n$ are the coefficients (model weightings for each feature) and β_0 is the model intercept. [64].

The model's parameters are typically estimated using the Ordinary Least Squares (OLS) method, which minimizes the Mean Squared Error (MSE) between the predicted and actual values:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{3.9}$$

Where n is the number of observations, y_i are the actual values, and $\hat{y}_i$ are the predicted values.

Linear regression's simplicity and interpretability make it a popular choice for many applications, but it has limitations. The model assumes a linear relationship between variables, which isn't always accurate. For example, compound interest grows exponentially rather than linearly because it accumulates on both the principal and previously earned interest. Additionally, linear regression is sensitive to outliers and can be affected by multicollinearity among independent variables [80].

To address some of these limitations, regularisation techniques have been developed. Ridge regression (L2 regularisation) adds a penalty term proportional to the sum of squared coefficients [70], while Lasso regression (L1 regularisation) uses the sum of absolute values of coefficients [186]. These methods help prevent overfitting and can improve the model's generalisation capabilities.

3.3.2 Logistic Regression

Logistic regression, developed in the mid-20th century [64], extends the principles of linear regression (a specialized form of generalised linear models) to binary classification tasks. Input features are mapped to probabilities using the logistic function, also known as the sigmoid function:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{3.10}$$

for the independent variable x. A logistic regression model is defined as:

$$P(Y = 1 \mid \mathbf{x}) = \sigma(\mathbf{x}^T \boldsymbol{\beta} + \beta_0) \tag{3.11}$$

Here, $P(Y = 1 \mid \mathbf{x})$ represents the probability of the binary outcome Y being 1 given the input features $\mathbf{x}$. The vector of input features is denoted as $\mathbf{x}$, the vector of model coefficients as $\boldsymbol{\beta}$, and the intercept term as β_0. The logistic function, $\sigma(\cdot)$, is defined in Equation 3.10.

The coefficients (β) in logistic regression have a meaningful interpretation. For a given feature, the exponential of its coefficient (e^{β}) represents the change in odds of the outcome for a one-unit increase in that feature, assuming all other features remain constant. A positive coefficient indicates that an increase in the feature is associated with an increase in the probability of the positive class, while a negative coefficient indicates the opposite.

The coefficients $\boldsymbol{\beta}$ are estimated using the method of Maximum Likelihood Estimation (MLE). In this approach, the parameter values that maximize the likelihood function, representing the probability of observing the given set of data, are found:

$$L(\boldsymbol{\beta}) = p(\mathbf{y} \mid \boldsymbol{\beta}) = \prod_{i=1}^{N} p(\mathbf{x}_i)^{y_i} (1 - p(\mathbf{x}_i))^{1-y_i} \tag{3.12}$$

where $p(\mathbf{x}_i) = \sigma(\mathbf{x}_i^T\boldsymbol{\beta}+\beta_0)$, $\sigma(z) = \frac{1}{1+e^{-z}}$, y_i is the observed binary outcome for the i^{th} sample, and $\mathbf{x}_i$ is the feature vector for the i^{th} sample. The log-likelihood, which is maximised in practice, is given by:

$$\log L(\boldsymbol{\beta}) = \sum_{i=1}^{n} \Big[y_i \log \sigma(\mathbf{x}_i^\top \boldsymbol{\beta}) + (1 - y_i) \log(1 - \sigma(\mathbf{x}_i^\top \boldsymbol{\beta})) \Big] \tag{3.13}$$

The performance of the logistic regression model is quantified using the Cross-Entropy Loss function, also known as Log Loss. This function measures the difference between the true labels and the predicted probabilities, and is defined as:

$$\text{Cross-Entropy Loss} = -\frac{1}{n} \sum_{i=1}^{n} [y_i \log p(\mathbf{x}_i) + (1 - y_i) \log(1 - p(\mathbf{x}_i))] \tag{3.14}$$

Minimisation of this loss function during training aligns with the maximisation of the log-likelihood, ensuring that the model's predicted probabilities closely match the actual data distribution.

As in linear regression, it is assumed that there is a linear relationship between the features and the log-odds of the outcome. Additionally, L1 and L2 regularisation techniques can be used to prevent overfitting.

3.3.2.1 Decision Trees

Conceptually, decision trees are among the simpler techniques for modelling non-linear distributions, in contrast to the linear models discussed previously. These hierarchical models partition data using conditional $if \ldots then \ldots else$ rules, consiting of:

1. **Root node**: The initial split of the dataset.
2. **Internal nodes**: Subsequent decision points.
3. **Leaf nodes**: Terminal nodes with final predictions.

Figure 3.10 illustrates this structure in regards to PD classification. The root node (UPDRS score) brances into internal nodes (Tremor Severity, Age), leading to leaf nodes with final classifications (Positive/Negative). This structure allows for intuitive interpretation of the model's decision-making process, tracing logic from root to leaf.

The tree's depth (longest root-to-leaf path) and breadth (nodes per level) influence mode complexity and its capacity to capture non-linear patterns in the data.

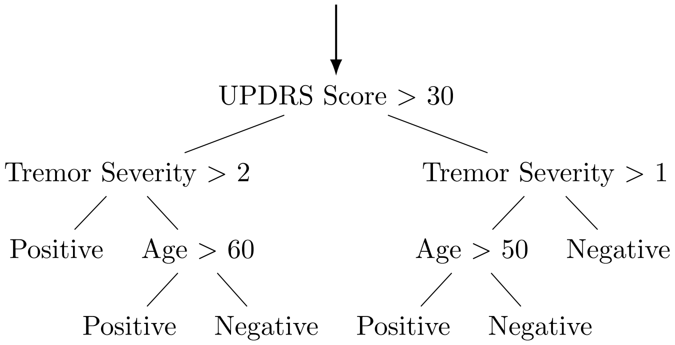


Figure 3.10: Illustrative decision tree model for Parkinson's classification based on UPDRS score, tremor severity, and age.

The construction of a decision tree involves selecting the best feature and threshold for splitting the data at each node. This selection is guided by impurity measures, which quantify the homogeneity of the target variable within the subsets. Common impurity measures include:

- **Entropy and Information Gain**: A measure developed by Claude Shannon [169], entropy quantifies the average "surprise" or uncertainty in the data. For binary classification, it's calculated as $H = -p_1 log_2(p_1) - p_2 log_2(p_2)$, where p_1 and p_2 are class probabilities. This sum represents the weighted average of information content for each class. Lower entropy is optimal, indicating that the subsets each have similar class labels. Conversley, information gain, calculated as $1 - Entropy$, measures the reduction in uncertainty achieved by a split. The Iterative Dichotomiser 3 (ID3) algorithm, introduced by Quinlan in 1986 [150], pioneered the use of information gain as a splitting criterion. However, it tends to favour features with many unique values, leading to the development of C4.5.

- **Gain Ratio**: To address the limitation of the ID3 algorithm, Quinlan's C4.5 [163] algorithm uses gain ratio instead of information gain. Gain ratio normalises information gain by the entropy of the feature itself, reducing bias towards multi-valued features.

- **Gini index**: Measures the probability of misclassification if an item were randomly labelled according to the class distribution in the subset. For binary classification, $Gini = 1 - (p_1^2 + p_2^2)$, where p_1 and p_2 are the proportions of each class. A Gini index of 0 represents perfect purity, while 1 indicates maximum impurity. The Classification and Regression Trees (CART) algorithm, introduced by Breiman et al. in 1984 [17], uses the Gini index for classification tasks.

The aim is to maximise the reduction in impurity (or information gain) with each split (vice versa for entropy), recursively building the tree until a stopping criterion is met.

Decision trees, despite their interpretability and versatility, face several key limitations:

- **Overfitting**: Deep trees often capture noise in training data, leading to poor generalization. Pruning techniques, such as those implemented in C4.5 and CART (cost-complexity pruning), help mitigate this issue. However, finding the optimal tree size remains challenging.
- **Instability**: Small variations in training data can result in significantly different tree structures, affecting model interpretation and consistency. This makes decision trees sensitive to outliers and noisy data.
- **Feature bias**: Trees tend to favor features with more levels or continuous variables, potentially leading to suboptimal splits. While algorithms like C4.5 address this to some extent with gain ratio, the issue persists in many implementations.
- **Limited decision boundaries**: Trees struggle with capturing complex, non-axis-parallel decision boundaries, often resulting in step-like decision surfaces that may not accurately represent the underlying data distribution.
- **Computational complexity**: For large datasets, considering all possible splits during tree construction can be computationally expensive.
- **Imbalanced data handling**: Trees may not perform well on imbalanced datasets, tending to favor the majority class. Techniques like weighted classes or sampling methods are often necessary to address this limitation.

These limitations have driven the development of ensemble methods like Random Forests and boosting algorithms, which aim to overcome some of these challenges while retaining the interpretability and flexibility of decision trees.

3.3.2.2 Random Forest

Random Forests, introduced by Breiman in 2001 [16], are an ensemble learning method that addresses many limitations of individual decision trees. This algorithm constructs multiple decision trees and combines their outputs to improve prediction accuracy and model robustness.

The core principles of Random Forests are bagging (bootstrap aggregating) and feature randomness. In bagging, each tree is trained on a random subset of the training data, sampled with replacement. This process creates diverse trees that capture different aspects of the data. Feature randomness further enhances diversity by considering only a random subset of features at each node split, typically $\sqrt{p}$ features for classification or $p/3$ for regression, where p is the total number of features.

Predictions in Random Forests are made by aggregating results from all trees. For classification tasks, a majority vote is used, while for regression, the average prediction is taken. This ensemble approach significantly reduces overfitting and improves generalisation compared to single decision trees.

Despite these benefits, Random Forests have limitations. They are less interpretable than single decision trees and can be computationally intensive, especially for large datasets.

3.4 Artificial Neural Networks

An Artificial Neural Network (ANN) consists of a network of interconnected nodes organised in layers, inspired by the function and structure of biological neural networks in the brain. The Perceptron, which models a single neuron's behaviour, serves as a foundational building block of ANNs.

The Perceptron model consists of several key components, illustrated in Figure 3.11. It receives inputs $(x_1, x_2, \ldots, x_n)$, analogous to dendrites receiving synaptic signals from other neurons. These inputs are associated with weights $(w_1, w_2, \ldots, w_n)$, representing the strength of connections between neurons. A summation function computes the weighted sum of these inputs. The result is then passed through an activation function (f), which determines the neuron's output, similar to an axon firing in a biological neuron. A typical activation function used in early perceptrons was the step function, which outputs 1 if the input is above a certain threshold, and 0 otherwise. A bias term (b_0 or w_0) is used to adjust the activation threshold. Mathematically, the perceptron's output is expressed as:

$$\hat{y} = f(b_0 + \mathbf{w}^T \mathbf{x}) \tag{3.15}$$

Where f is the activation function, b_0 is the bias, $\mathbf{w}$ is the weight vector, and $\mathbf{x}$ is the input vector.

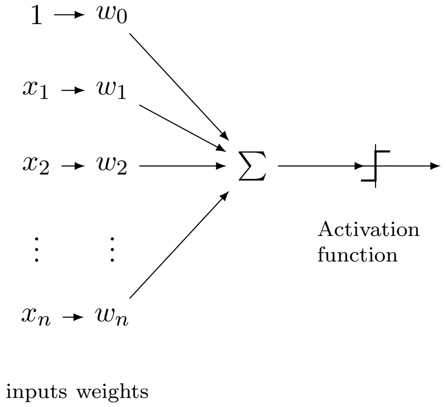


Figure 3.11: Structure of a perceptron, showing inputs, weights, summation, and activation function.

The concept of the perceptron was introduced by Frank Rosenblatt in 1958 [157]. It builds upon the earlier work of McCulloch and Pitts [110], who proposed a simplified neuron model with weighted binary inputs and produces a binary output based on an adjustable threshold value. Rosenblatt also developed a supervised learning algorithm for binary classification using a single layer of perceptrons, the simplest feed-forward neural network. This algorithm, inspired by Hebbian learning ("neurons that fire together, wire together"), iteratively adjusts the weights of a perceptron to minimize classification errors. It updates the weights based on the difference between predicted and actual outputs, but crucially, only when a misclassification occurs. For each misclassified data point, the algorithm modifies the weights in a way that reduces the error, refining the model's accuracy over time. Throughout the 1960s, Widrow and Hoff made several developments [196, 195], including: the Least Mean Squares (LMS) algorithm for updating weights and the Adaptive Linear Nueron (ADALINE) .

Despite initial enthusiasm, ANN research stagnated following the 1969 publication of Minsky and Papert's book "Perceptrons" [113], which demonstrated that single-layer networks could not solve non-linearly separable problems (e.g., the XOR function). This limitation highlighted the need for more complex architectures, which were limited by the computational performance of the time. Accordingly, the dramatic increase of available computational power has been accompanied by an rise in NN usage. This recently found popularity has given rise to many new NN models being proposed, most notably including the invent of deep learning.

In modern machine learning, "deep learning" refers to a range of neural network architectures with multiple hidden layers (layers that sit between the input and output layers). These include Multi-Layered Perceptrons, Convolutional Neural Networks, Recurrent Neural Networks, and Transformers. These architectures have demonstrated remarkable success across various domains, from image and speech recognition to natural language processing and beyond, solidifying ANNs as a cornerstone of contemporary artificial intelligence research and applications. This thesis uses models based on Convolutional Neural Networks but [165] provides and overview of a broad range of current NN architectures.

3.4.1 Deep Neural Networks

The Multilayer Perceptron (MLP) architecture extends the basic perceptron model by stacking multiple layers of interconnected perceptrons [141]. This architecture, combined with non-linear activation functions, allows MLPs to learn and model complex non-linear relationships in data, overcoming the limitations of single layer perceptrons highlighted by Minsky and Papert [113].

An MLP model consists of three structural components, illustrated in Figure 3.12. The first is the input layer which receives the initial data features. The last is the output layer, which produces the final classification. Nested between the input layer and output layer are one or more intermediate layers, collectively referred to as the hidden layers, the number of which is the model's depth. Networks with multiple hidden layers are considered "deep" and form the foundation of deep learning. The popularity and success of deep learning can be attributed in part to an unexpected phenomenon: the remarkable improvement in performance as networks grow deeper. This scaling effect has led to deep learning models becoming state-of-the-art in numerous fields. However, exploring and leveraging this relationship between depth and performance has only become feasible with the advent of modern computational resources, particularly Graphics Processing Units (GPUs).

In a fully connected MLP model, each perceptron in each layer is connected to every perceptron in the following layer (See Figure 3.12). As information propagates through an MLP, each successive hidden layer is thought to abstract and transform features from the previous layer. This hierarchical feature extraction allows the network to learn increasingly complex representations of the input data. However, it's worth noting that the exact nature of these abstractions can be difficult to interpret, especially in very deep networks, making them a 'black box'.

While fully connected networks are powerful, they can be prone to overfitting, especially when dealing with limited training data. To address this, various regularisation techniques have been developed. One such technique, particularly relevant to the network structure, is dropout [177]. Dropout is a regularisation method that randomly "drops out" (i.e., temporarily removes) a proportion of neurons and their connections during training. This process can be visualised as temporarily creating a sparser version of the network for each training iteration. Dropout helps prevent co-adaptation of neurons and reduces overfitting, often leading to better generalisation. During inference, all neurons are used, but their outputs are scaled to compensate for the higher number of active units compared to training.

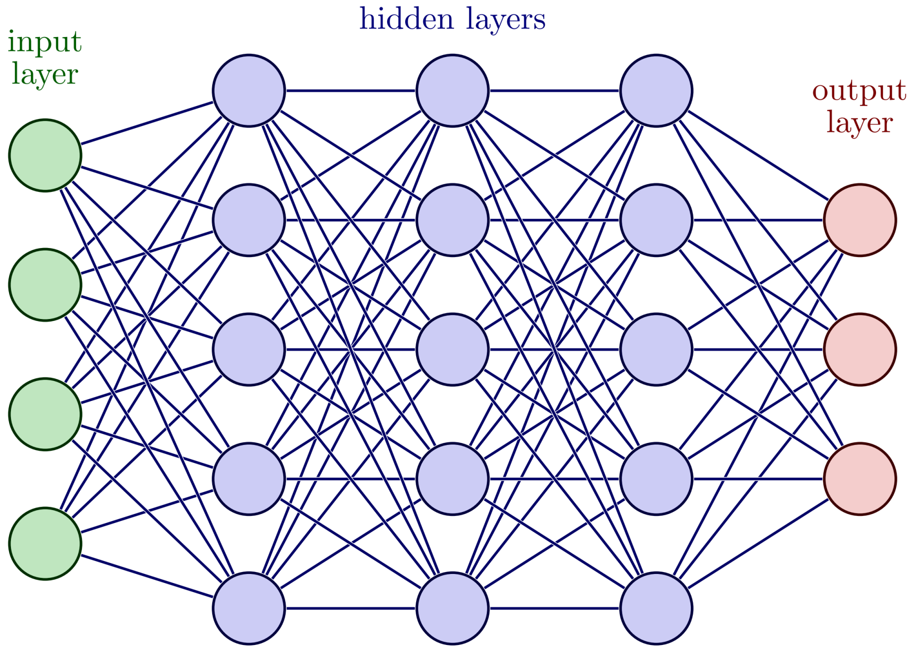


Figure 3.12: MLP network, adapted from [125]

Backpropagation and gradient descent form the foundation of neural network training [161]. Backpropagation efficiently computes gradients of the loss function with respect to all network weights by leveraging the chain rule and reusing intermediate calculations as it propagates error backwards through the network. This process involves a forward pass to compute activations and loss, followed by a backward pass to calculate gradients. Gradient descent then uses these gradients to iteratively update weights, aiming to minimize the loss function. The basic weight update equation is:

$$W' = W - \eta \nabla J(W)$$

Where $W^{'}$ represents the updated weights, W represents the current weights, η (eta) is the learning rate, ΔW denotes the gradient with respect to W, and $J(W)$ is the loss function. The learning rate η is a crucial hyperparameter controlling the step size at each iteration. This update process is repeated for multiple passes through the entire dataset, with each complete pass called an epoch. The number of epochs is another important hyperparameter, as too few may result in underfitting, while too many can lead to overfitting. Advanced optimisers like Adam [88] adapt learning rates for each parameter, and techniques like the one-cycle policy [174] (used in the fast.ai library employed in this thesis) dynamically adjust the learning rate during training. While classic gradient descent updates weights based on the entire dataset per epoch, stochastic gradient descent [87] uses mini-batches for improved efficiency and noise injection within each epoch. The interplay between backpropagation and these optimisation techniques enables neural networks to learn complex, non-linear mappings from input to output across multiple epochs.

A constraint when using gradient descent is that activation functions must be differentiable. The following are some of the common activation functions used in deep learning:

- **Hyperbolic Tangent Function (tanh)**: Defined by $\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$, tanh outputs values in the range $[-1, 1]$. Similar to the sigmoid but with a broader output range, it also suffers from vanishing gradient problems in deep networks.
- **Rectified Linear Unit (ReLU)**: Introduced in [122], given by $f(x) = \max(0, x)$, ReLU is computationally efficient and serves as the default activation function for many types of neural networks. However, it is sensitive to outliers and can suffer from "dying ReLU" problem [105], where neurons never activate.
- **Leaky Rectified Linear Unit (Leaky ReLU)**: Introduced in ,a variant of ReLU, defined as $f(x) = \max(\alpha x, x)$, where α is a small constant. This function aims to solve the "dying ReLU" problem by allowing a small gradient when $x < 0$.
- **Softmax**: Often used in the output layer of a classifier to represent probabilities, Softmax normalises the input into a probability distribution over multiple classes. Mathematically, given input vector $x \in \mathbb{R}^K$, the Softmax function is defined as $S(x)_j = \frac{e^{x_j}}{\sum_{k=1}^{K} e^{x_k}}$ for $j = 1, ..., K$.

The efficacy of gradient descent and backpropagation heavily relies on the choice of activation functions used in the neural network. Activation functions introduce non-linearity into the network, allowing it to learn complex patterns. Crucially, these functions must be differentiable to enable the calculation of gradients during backpropagation.

Early neural networks often used sigmoid or hyperbolic tangent (tanh) functions. However, these functions can lead to the vanishing gradient problem in deep networks, where gradients become extremely small as they propagate backwards, slowing down or halting learning. This issue arises because the derivatives of these functions approach zero for very large or small inputs.

To address this, modern neural networks often employ the ReLU activation function, defined as f(x) = max(0, x). ReLU and its variants (such as Leaky ReLU) have several advantages. First they are computationally efficient. ReLU has no upper bound for positive inputs like sigmoid or tanh functions do. As a result, the gradient is not reduced to near-zero values during backpropagation. With ReLU, perceptrons output zero for all negative inputs, leading to sparse activations in the network. Sparse activations limit the number of pathways through which gradients can flow. This means that only a subset of perceptrons are active at any given time, leading to less complex gradient updates, and reduces the risk of large, unstable gradients that can cause exploding gradients. However with this, ReLU can suffer from the ”dying ReLU” problem, where neurons can get stuck in an inactive state. This has led to the development of variants like Leaky ReLU [106] and Parametric ReLU [66].

3.5 Convolutional Neural Networks

3.5.1 Convolution operator and kernels as feature detectors

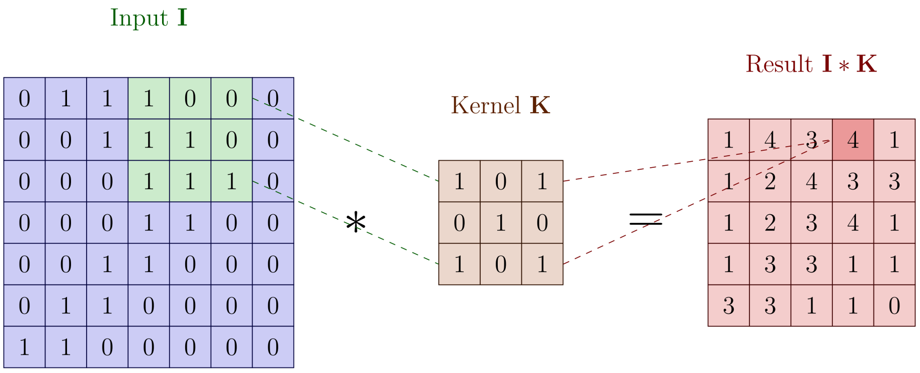
In digital image processing, the discrete convolution operator is a mathematical approach that can emphasise specific features within an image, such as edges, textures and patterns. As such the output of the operation is also an image (often referred to as a feature map). The discrete convolution operator for 2D images can be defined as:

$$(I * K)(i,j) = \sum_{m}\sum_{n} I(i-m, j-n) \cdot K(m,n)$$

Where i and j are the co-ordinates of the output feature map, and m and n are the co-ordinates within the kernel. A convolution operation is identical to determining the cross-correlation between a signal and a kernel except for that in convolution the kernel has been time-reversed. The kernel is generally a small matrix that slides over the input matrix, performing element-wise multiplication and summation to produce each element of the output. Figure 3.13, illustrates this.

The Sobel filter [175] is a type of image gradient operator used to detect edges in images. It consists of two 3x3 kernels, which are convolved with the input image to compute approximations of the derivatives in the vertical and horizontal directions.

The Sobel Vertical Kernel (G_x) detects horizontal edges by highlighting vertical changes in



$$\begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 4 & 3 & 4 & 1 \\ 1 & 2 & 4 & 3 & 3 \\ 1 & 2 & 3 & 4 & 1 \\ 1 & 3 & 3 & 1 & 1 \\ 3 & 3 & 1 & 1 & 0 \end{bmatrix}$$

Figure 3.13: The input matrix I is convolved with the kernel K to produce the result matrix I * K. The highlighted cells in the input show the current receptive field of the kernel, and the corresponding output in the result matrix is highlighted. Adapted from [124]

intensity and is given by:

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \tag{3.16}$$

Similarly, the Sobel Horizontal Kernel (G_x) detects vertical edges by highlighting horizontal changes in intensity:

$$G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \tag{3.17}$$

The results from both kernels can be combined to produce a gradient magnitudes image, revealing edges in all directions.

$$G = \sqrt{G_x^2 + G_y^2} \tag{3.18}$$

This can also be done for the orientation of the gradients.

In CNNs, the network learns its own kernels during training, rather than using predefined kernels like the Sobel filter. These learned kernels can detect a wide variety of features, from simple edges and textures in early layers to more complex patterns in deeper layers. In [91], the authors demonstrated that kernels in the first layer learned to detect edges and

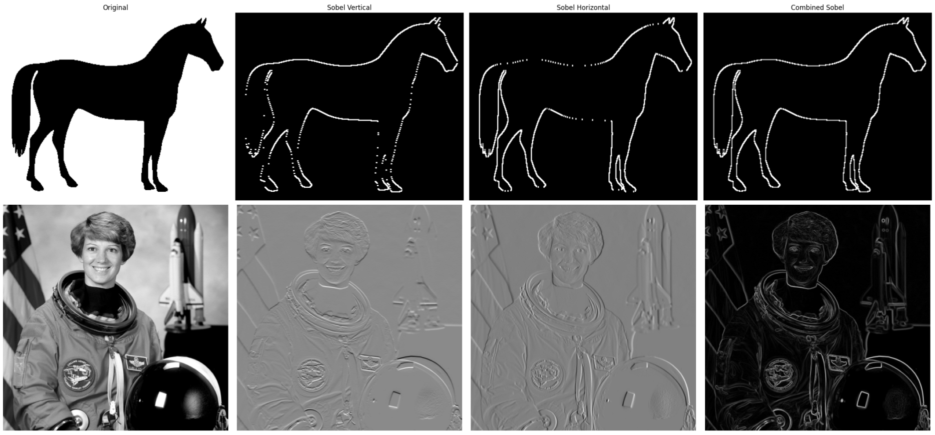


Figure 3.14: Demonstration of edge detection using the Sobel operator applied to binary (top) and greyscale (bottom) images. Note that the binary image maintains a gradient of 1, while in the greyscale image, the magnitude varys in intensity. From left to right the columns illustrate: (1) The original image, (2) The feature map resulting from the G_x convolution, (3) The feature map resulting from the G_y convolution, (4) Magnitude of G_x and G_y feature maps.

color blobs, as illustrated in Figure 3.15. This highlights a fundamental difference between convolutions in traditional image processing and those in CNNs. Mathematically, in image processing, operations like Gaussian blur are typically applied to each channel of an RGB image independently and then concatenated. In CNNs, kernels operate across all input channels simultaneously, computing a sum over all channels for each spatial position. This multi-channel convolution allows CNNs to capture inter-channel relationships, enabling the detection of more complex features that combine information across channels. Conceptually, this means CNN kernels can learn to respond to specific color-edge combinations or other cross-channel patterns, rather than treating each color channel in isolation.

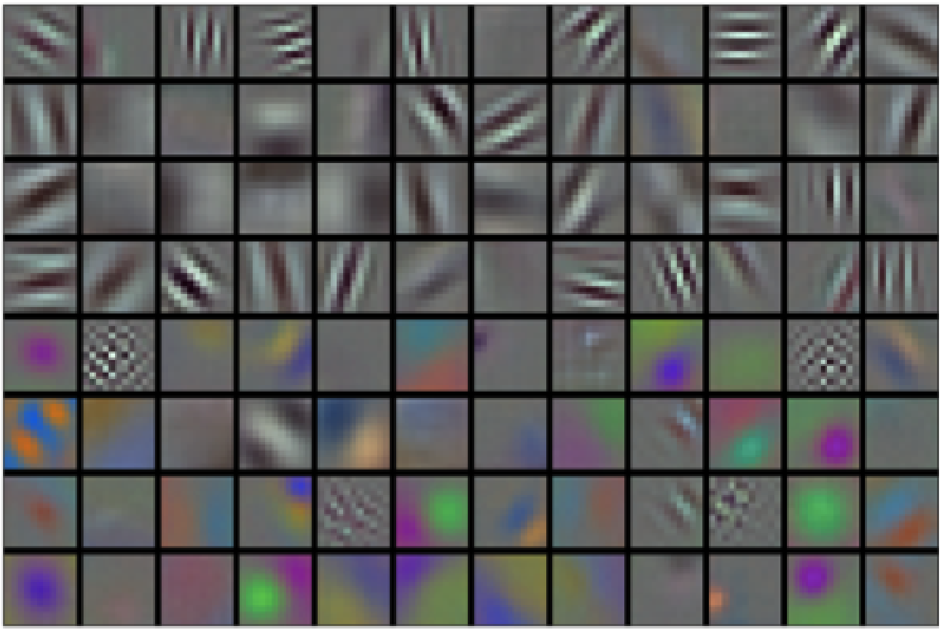


Figure 3.15: Visualisations of convolutional filters in the first layer of AlexNet [91].

3.5.2 Architecture

A standard CNN architecture comprises an input layer, convolutional layers, activation layers, pooling layers, fully connected layers, and an output layer [204]. Each type of layer serves a specific purpose in feature extraction and transformation. This standard architecture is illustrated in Figure 3.16.

The input to a CNN is typically a multi-dimensional tensor. For image data this is usually a 3D tensor with dimensions representing width, height, and depth (often corresponding to colour channels). This structure can be extended to handle more complex inputs, such as video data, which might include an additional time dimension [83].

The convolutional layers form the core of the CNN architecture. Each convolutional layer contains a set of learnable filters or kernels. The number of kernels in each layer is an architectural choice and can vary depending on the complexity of the task. The output of each convolutional layer is a set of feature maps. Each feature map is the result of applying a specific kernel across the entire input, capturing different aspects of the input data. As data progresses through the convolutional layers, there's often a reduction in spatial dimensions (width and height), while the depth (number of feature maps) typically increases. The same kernel is applied across the entire input, allowing for efficient parameter sharing and the ability to detect features regardless of their position in the input. Each neuron in a feature map is connected only to a local region of the input, rather than to all input neurons, significantly reducing the number of parameters compared to fully connected layers.

It's important to note that activation functions are typically applied after each convolutional operation. Immediately after the convolution operation, the activation function is applied element-wise to this pre-activation feature map. This means that the activation function is applied independently to each element of the feature map.

Pooling layers are often interspersed between convolutional layers. They help to reduce the spatial dimensions of the feature maps, making the network more computationally efficient and helping to achieve some degree of translational invariance.

The final part of the CNN architecture typically consists of one or more fully connected layers, essentially forming an MLP network. These layers take the high-level features extracted by the convolutional layers and use them for tasks such as classification. The number of neurons in the final layer usually corresponds to the number of classes in the classification task.

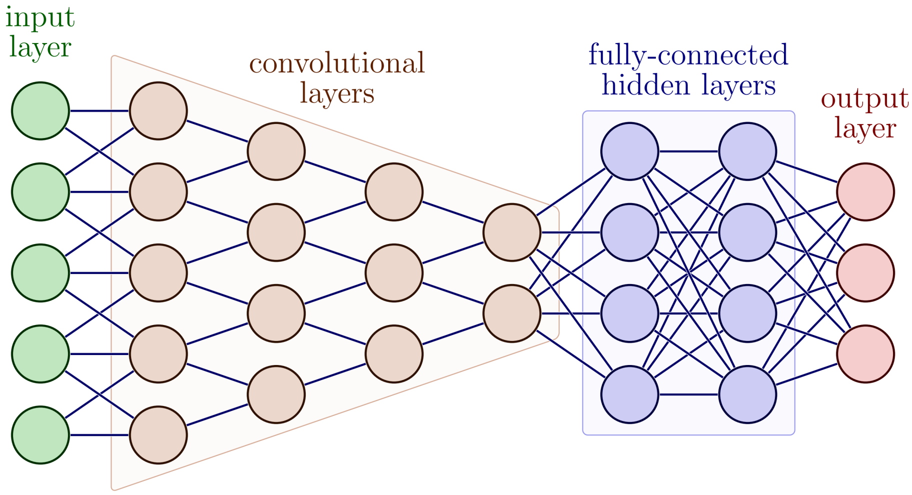


Figure 3.16: CNN Architecture, adapted from [125]

3.6 Deep Learning Based Semantic Segmentation

3.6.1 U-Net

The U-net architecture is a deep learning model that was proposed by Ronneberger et al. in 2015 [156]. It was specifically designed for semantic segmentation tasks, where the goal is to assign a label to each pixel in an image. The U-net architecture falls under the class of Fully Convolutional Networks (FCNs). It consists of two main parts, an encoder and a decoder. The encoder is responsible for capturing the high-level features of the input image, while the decoder is responsible for generating the segmentation map. The U-Net architecture incorporates skip connections between the encoder and the decoder, allowing for more precise localization by concatenating high-resolution features from the encoder with upsampled output from the decoder. This is particularly beneficial for tasks where the spatial context is crucial. Additionally, the U-Net model is often enhanced by employing Residual Networks (ResNets) as a backbone, which allows for the training of deeper networks by mitigating the vanishing gradient problem [65]. Utilising pretrained ResNets can further improve the model's performance through transfer learning, where knowledge gained from one task is applied to another. This is particularly useful when labeled data may be scarce.

3.7 Convolutional Based Time Series Classification Algorithms

Multivariate time series classification models deal with time series data where multiple variables or channels are observed at each time point. Each channel may represent different features or sensors. These models are designed to classify sequences into predefined classes

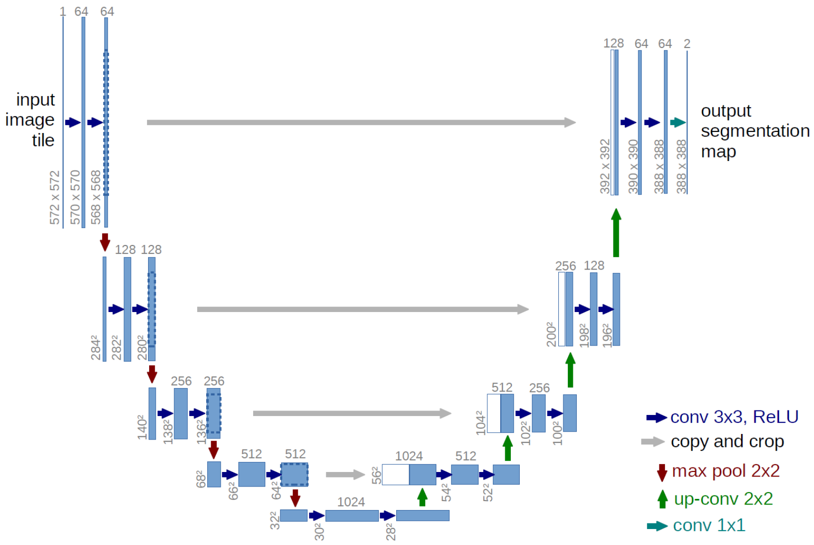


Figure 3.17: U-Net architecture [156]

based on temporal patterns across multiple channels. A recent review on multivariate time-series classification approaches recommended InceptionTime, and Rocket [160].

3.7.1 InceptionTime

InceptionTime is a specialised deep learning architecture tailored for time series classification. The model was proposed by Ismail Fawaz et al. in 2020 and employs an ensemble of five deep learning models [78]. As illustrated in Figure 3.18, each module is organized into blocks, which themselves are composed of Inception modules (shown in Figure 3.19). These Inception modules make use of one-dimensional convolutions to capture complex temporal features at various scales. The architecture also incorporates batch normalisation and residual connections to expedite training convergence and improve generalisation performance.

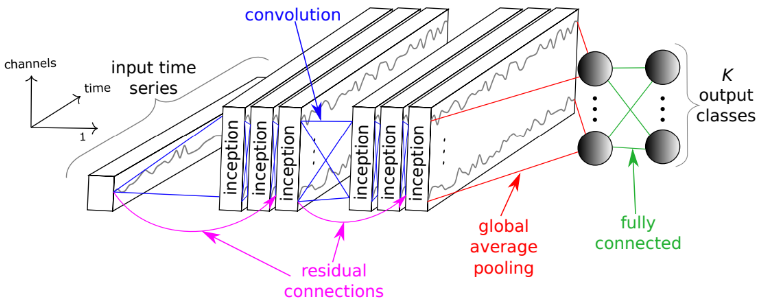


Figure 3.18: A diagram of the Inception network [78]

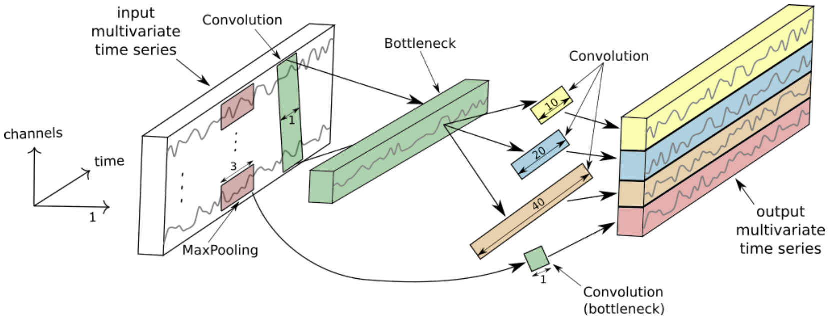


Figure 3.19: A diagram of the Inception module [78]

3.7.2 Rocket

RandOm Convolutional KErnel Transform (Rocket) introduced in 2019 [44], represents a contrasting approach to time series classification. Unlike InceptionTime, Rocket is not inherently a machine learning model. It is a feature extraction method that employs a large number of randomly generated kernels to transform the original time series data. Subsequently, a linear classifier is used for the actual classification task. The method is computationally efficient and can be paired with a variety of classifiers, such as Logistic regression or Ridge regression.

MiniRocket is a streamlined variant of Rocket introduced a year later in 2020 [45], designed for even faster feature extraction while maintaining comparable accuracy. Given its computational efficiency and performance, the authors suggest it as the default variant of Rocket.

MultiRocket, introduced in 2022 [182], extends the capabilities of MiniRocket by incorporating multiple pooling operators and transformations to diversify the generated features. Specifically, it applies first-order differences to transform the original series and utilises convolutions on both the raw and transformed series. Four distinct pooling operators are then applied to the outputs of these convolutions, enhancing the robustness and expressiveness of the feature set.

Chapter 4

Diagnosing Parkinson's Disease and Clinically Slight Bradykinesia From Raw Positional Data

PD is typically diagnosed and monitored through assessments of motor function, which include upper limb tasks such as finger tapping, hand opening/closing, and pronation-supination. While these evaluations are well-established, they rely on visual assessment, leading to scoring criteria that are necessarily broad to account for the inherent limitations of human observation. As a result, these tasks are challenging to judge consistently, even for expert clinicians, who often only achieve moderate inter-rater agreement.

The need for a more objective, consistent, and data-driven approach to assessing motor impairment has led to growing interest in the application of machine learning models to complement traditional clinical assessments. This chapter focuses on recent advancements in multivariate time-series classification models to address two key questions: first, whether these models can accurately replicate clinician judgments by distinguishing between normal and impaired performance; and second, whether they can differentiate between individuals with PD and healthy controls, even in cases where visible impairment is subtle or absent.

To investigate these questions, the performance of state-of-the-art end-to-end time-series classification models—InceptionTime [78], MiniRocket [45], and MultiRocket [182] are evaluated in diagnosing PD and identifying subtle motor impairments during these upper limb assessments. These models offer the potential to go beyond traditional feature engineering by directly learning from raw positional data, identifying patterns and subtleties that may not have been captured or explicitly defined in previous methods. This capability is particularly relevant given evidence that expert clinicians often recognise features or cues beyond established scoring systems, suggesting that valuable information remains unquantified. Although this study does not include detailed explainability analyses, it represents an initial exploration into the application of these models.

4.1 Background

The measurement of hand kinematics in motor tasks typically involves tracking key points on the subject's hand. This can be achieved using either wearable tracking sensors or markerless pose estimation from video. Wearable sensors, such as accelerometers, gyroscopes, and electromagnetic trackers, have been widely utilised for this purpose. Several studies have explored the application of these sensors in hand tracking for PD movement analysis [178, 94, 56, 120].

In recent years, markerless methods using consumer-grade devices have gained popularity due to their low cost and ease of use. The Leap Motion controller, for instance, has been employed in several studies exploring hand tracking in the context of PD [19, 117, 90, 20, 53]. Additionally, advancements in computer vision driven by deep learning have enabled software solutions like DeepLabCut [109] to perform precise pose estimation, as demonstrated in recent work [114].

The conventional approach to applying machine learning in these assessments involves first extracting features from the raw recordings. Typically, this process begins by reducing the dimensionality of the hand key points into a one-dimensional time series. For instance, finger tapping is often represented by the amplitude between the index finger and thumb. Feature extraction is then conducted in two stages: first, by calculating statistics for each cycle (e.g., each tap), and then by aggregating these statistics into a single score for the entire recording.

In their work, the authors of [114] group features based on clinically relevant aspects of the signal, such as speed, amplitude, hesitations, and fatigue (e.g., decrementing amplitude). Common cycle-level statistics include maximum and minimum amplitude, velocity, and acceleration, with additional segmentation by phases of the movement, such as opening velocity [200]. These features are typically aggregated using metrics such as mean, range, and coefficient of variation, while for fatigue assessment, the slope of the regression line through peak amplitudes is often used. It is important to reiterate that these studies apply machine learning models such as (SVM's [19], Random Forest [114], Logistic Regression [200]), after the signal has been first reduced and constrained to researcher defined measures from a 1D signal.

While this feature-based approach has proven effective, it may overlook more complex patterns in the data. A study by Williams et al. [197], highlighted this limitation, revealing that expert clinicians' overall impressions of PD in finger tapping videos were not strictly aligned with formal bradykinesia criteria. This discrepancy suggests the existence of subtle, yet unquantified, patterns that experienced clinicians intuitively recognise, underscoring the need for more comprehensive measures of movement impairment in PD.

Given these challenges, recent research has focused on end-to-end machine learning approaches that can learn directly from raw data without extensive feature engineering. These methods, including convolutional neural networks [130, 199], echo-state networks [94], and Cartesian Genetic Programming (CGP) [120, 56], offer the potential to capture more nuanced aspects of movement impairment.

The present work builds upon these advancements by exploring models from the field of Time Series Classification (TSC). Specifically, InceptionTime [78], MiniRocket [46], and MultiRocket [182] have been selected for their strong performance on limited datasets [160], a common constraint in clinical research. By applying these models to upper limb assessments in PD, this study aims to evaluate their effectiveness in distinguishing between PD patients and controls, as well as identifying subtle motor impairments that may be overlooked in traditional assessments.

4.2 Data Collection

4.2.1 Subjects

The dataset used in this study originates from the clinical caseload managed by Dr. Gao at Ruijin Hospital. The dataset encompasses 148 individuals diagnosed with PD, and a control group of 47 age and gender matched Healthy Controls (HCs) (or Normal Controls (NCs)), who were recruited from patient's spouses and companions. It should be noted that the presence of the swallow tail sign [15] was a supporting feature in all PD patients. All participants are right-hand dominant, the selection process did not apply exclusion criteria based on medication status (either OFF or ON) or cognitive condition, given that these aspects were not the focal point of investigation. All procedural frameworks of the study secured approval from the Ethics commitee of Ruijin Hospital, affiliated with the Shanghai

Jiao Tong University School of Medicine. Furthermore, all participants engaged in the study engaged in their consent through written agreements, adhering to ethical research guidelines.

	NC	PD
Age, years	59.1 ± 6.3, (47.0-70.0)	59.9 ± 10.2, (35.0-80.0)
Disease duration, years	-	3.1 ± 2.8, $(0.08 - 20.0)$
Hoehn-Yahr stage	-	1.83 ± 0.57, $(1, 3)$
MMSE	28.9 ± 0.9, (27.0-30.0)	25.9 ± 4.1, (6.0-30.0)
Gender, M:F	4:5	16:21
Number of subjects	47	148

Table 4.1: Demographics and clinical characteristics of the study participants, with mean, standard deviation and range.

4.2.2 Protocol

The dataset consists of three exercises for the following MDS-UPDRS [61] tasks:

- **3.4 Finger tapping**: The patient is instructed to tap the index finger on the thumb ten times as quickly and as big as possible.
- **3.5 Hand movements**: The patient is instructed to open and close the hand ten times as quickly and as big as possible.
- **3.6 Pronation-Supination movement of the hands**: The patient is instructed to extend the arm out in front of their body with the palms down, and then to turn the palm up and down alternately ten times as fast and fully as possible.

Each task was scored on a scale from 0 to 4 using the MDS-UPDRS scoring criteria, with higher scores indicating increased severity. Each of the tasks contain similar marking criteria, which is provided in Table 4.2. Visual representations of the tasks are provided in Figure 4.1.

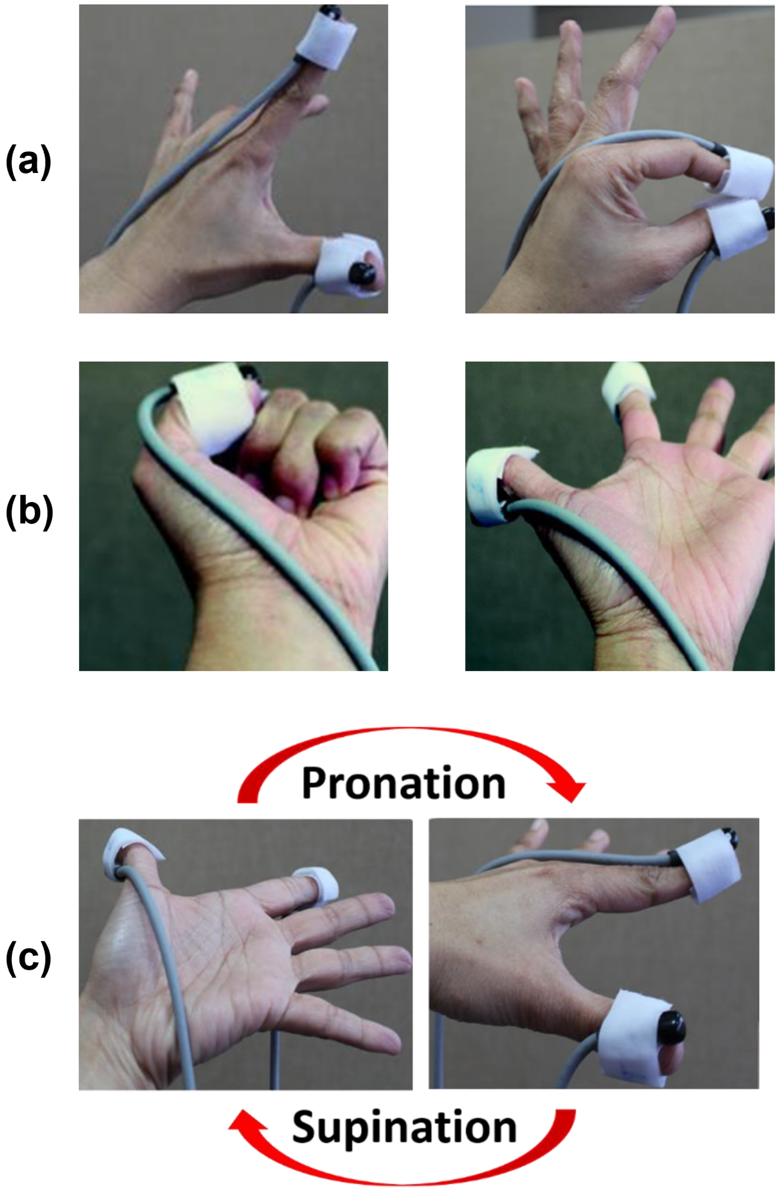


Figure 4.1: Each row depicts the following tasks, (a) finger tapping, (b) hand open and close, (c) pronation-supination movement of the hand. Each movement is periodic, oscillating between two states, for the duration of the assessment.

Score	Category	Description
0	Normal	No problems.
1	Slight	Any of the following: a) the regular rhythm is broken with one or two interruptions or hesitations of the movement; b) slight slowing; c) the amplitude decrements near the end of the task.
2	Mild	Any of the following: a) 3 to 5 interruptions during the movements; b) mild slowing; c) the amplitude decrements midway in the task.
3	Moderate	Any of the following: a) more than 5 interruptions during the movement or at least one longer arrest (freeze) in ongoing movement; b) moderate slowing; c) the amplitude decrements starting after the 1st sequence.
4	Severe	Cannot or can only barely perform the task because of slowing, interruptions, or decrements.

Table 4.2: MDS-UPDRS Task Scoring Criteria (slightly adapted to be more general for all three tasks) [61].

4.2.3 Equipment

Hand kinematics were recorded using the Polhemus Patriot M Electromagnetic (EM) tracking system [140], from which a pair of sensors were attached to the subjects index finger and thumb as shown in Figure 4.2. With the system, a source is placed in front of the subject that transmits an electromagnetic dipole field, each sensor contains three orthogonal receiver coils in which voltage is induced from the field, resulting in the measurements that are used to reconstruct the pose (six degrees of freedom) of the sensor. The position and orientation of each sensor is sampled at a frequency of 60 Hz. The accuracy of the sensor is within 1.5mm of Cartesian coordinates (x,y,z) and 0.4 degrees for orientation (yaw, pitch and roll)[1], providing that the distance between the transmitter and receiver is less than 76 cm. Only positional data is used in this study.

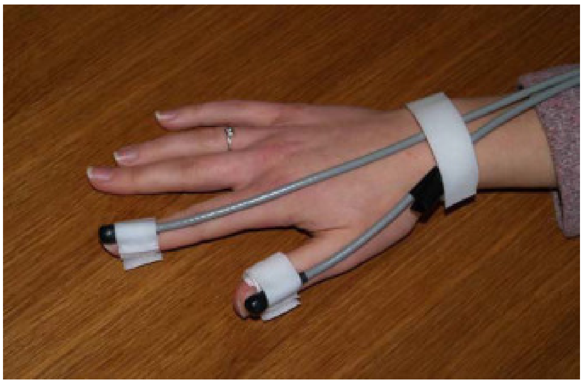


Figure 4.2: The positional sensors were attached on top of the nailbeds of the index finger and thumb.

This EM approach is free from alignment, lighting and occlusion issues common with optical approaches, and avoids drift errors typical in gyro, accelerometers and magnetometers. Nonetheless, it remains sensitive to disturbances to other magnetic fields or conductive materials.

4.3 Noise Analysis

An investigation was conducted to investigate the frequency content of the signals and the resulting impact of applying a smoothing filter in effort to reduce noise in the dataset.

[1] Intrinsic.

4.3.1 Raw Data

Let S be the set of sensors, where $S = \{1, 2\}$. Sensor 1 is placed on the index finger, and sensor 2 is placed on the thumb.

The positional coordinates of each sensor $s \in \{1, 2\}$ at timestamp t_i are given by $x_s(t_i)$, $y_s(t_i)$, and $z_s(t_i)$. These Cartesian coordinates represent the positions of the sensors in relation to the magnetic source. The units of measurement are in cm. Let $\mathbf{P}_s(t_i)$ be the positional vector of the s-th sensor at time t_i.

The orientation of each sensor $s \in \{1, 2\}$ at timestamp t_i is given by three Nautical Euler angles: azimuth $\alpha_s(t_i)$, elevation $\beta_s(t_i)$, and roll $\gamma_s(t_i)$. These angles are intrinsic rather than extrinsic, meaning that the angles are defined in terms of rotations around the sensor's local axes, rather than the axes of the magnetic source. The units of measurement are degrees.

4.3.2 Extracted Signals

For this analysis a uni-variate signal was extracted from the raw data for each task. This resulted in a signal $\mathbf{V} = (v_1, \ldots, v_i, \ldots, v_n)$, where v_i is the value of the i-th sample.

4.3.2.1 Finger Tapping and Hand Open-Close

Given the hand-key points available, the best uni-variate approximation of the movement is given by the Euclidean distance between the two sensors. This is typical for finger tapping, as the Euclidean distance can be used to find the amplitude of a tap [94, 19, 77, 21, 118, 117, 119, 164, 181].

Mathematically, the Euclidean distance between the two sensors at time t_i can be represented as the distance between $\mathbf{P}_1(t_i)$ and $\mathbf{P}_2(t_i)$, calculated as:

$$d(t_i) = \sqrt{(x_1(t_i) - x_2(t_i))^2 + (y_1(t_i) - y_2(t_i))^2 + (z_1(t_i) - z_2(t_i))^2}$$

or using positional vectors:

$$d(t_i) = \|\mathbf{P}_1(t_i) - \mathbf{P}_2(t_i)\|$$

4.3.2.2 Pronation-Supination

The movement involves repeated rotation between the wrist's pronated and supinated positions. The ideal placement for capturing this motion would be a sensor on the wrist. Since the sensors are placed on the index finger and thumb, the roll of the index finger ($\gamma_1(t_i)$) is used instead. This should be satisfactory, as the finger is stretched out and collinear with the wrist.

4.3.3 Deriving Frequency From Manual Annotation

The peaks of the extracted signals were manually annotated using a bespoke tool. This annotation tool is discussed in more detail in Appendix B; the peaks were also considered for a data driven approach to segmenting the signals. This resulted in a set of peak indices $P = \{p_1, p_2, \ldots, p_n\}$ for each recording. Thus, segments of the signal (e.g., a tap including both the closing phase and opening phase) were bounded by the following ($p_m : p_{m+1}$) for $m < n$ and begins at 1. This results in $n - 1$ segments per recording.

The frequency of movement from manual annotation (F_M) of a signal is given by the number of segments divided by the duration between the first and last peak:

$$F_M = \frac{n-1}{p_n - p_1} \cdot \left(\frac{1}{f_s}\right)$$

Where, f_s is the sampling rate.

Figure 4.3, shows the distribution of frequencies per task type. None of the recordings exceed 4 Hz.

4.3.4 Filter Selection

A fourth-order Butterworth [22] filter with a cutoff frequency of 5 Hz was selected as an appropriate filter. This choice was informed by several key considerations. Firstly, the cutoff

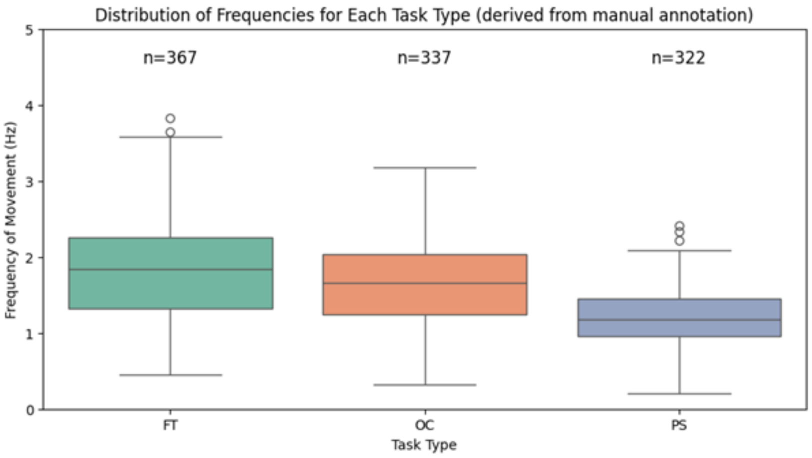


Figure 4.3: Distribution of frequencies per task type.

frequency was set at 5 Hz because the frequency of movement for all tasks remained below this threshold. This setting ensures the preservation of relevant movement data while effectively attenuating higher frequency noise. Secondly, the fourth-order filter was chosen for its roll-off steepness. While higher order filters can provide sharper cutoffs, a fourth-order filter offers a suitable balance for this application. Lastly, Butterworth filters are renowned for their maximally flat frequency response in the passband, a characteristic that aids in preserving the shape of signal components below the cutoff frequency.

The filtering process can be mathematically described as follows: Let $\mathbf{V} = [v_1, v_2, \ldots, v_n]$ be the original signal. The filtered signal $\mathbf{V}_f$ is obtained by applying the Butterworth filter to $\mathbf{V}$. In the frequency domain, the magnitude response of the Butterworth filter is given by:

$$|H(\omega)|^2 = \frac{1}{1 + \left(\frac{\omega}{\omega_c}\right)^{2N}}$$

Where ω is the angular frequency, ω_c is the cutoff frequency ($2\pi \times 5$ Hz) and N is the filter order (4).

The filtered signal $\mathbf{V}_f$ is obtained by convolving $\mathbf{V}$ with the impulse response of the Butterworth filter in the time domain.

4.3.5 Signal to Noise Ratio

4.3.5.1 Approach for Smoothing the Pronation-Supination Signal

The pronation-supination task, which measures rotational movement of the forearm, utilises Euler angles to represent orientation in three-dimensional space. However, these Euler angles present unique challenges in signal processing due to their inherent discontinuities. Specifically, when an angle transitions from 359° to 0° (or vice versa), it creates an artificial jump in the timeseries that does not reflect the true continuous nature of the rotational movement. This discontinuity can lead to significant errors in subsequent analyses if not properly addressed. To address these challenges, a specialised three-step approach was implemented:

1. **Angular Velocity Calculation**: Angular velocity was derived from the raw Euler angle data using a custom shortest-angle algorithm. This step is crucial because Euler angles are bounded within a range of 2π radians, leading to discontinuities when an angle exceeds this value (e.g., transitioning from 359° to 0°). The shortest-angle algorithm ensures that the true rotational motion is captured without introducing artificial jumps in the signal. It calculates the smallest angular difference between two angles, considering the circular nature of angular measurements. The core principle can be expressed mathematically as:

   $$\Delta\theta = ((\theta_2 - \theta_1 + 180^\circ) \mod 360^\circ) - 180^\circ$$

   Where θ_1 and θ_2 are the two angles being compared and, $\Delta\theta$ is the shortest angular difference.

   This process ensures that the calculated angular difference always represents the shortest path between the two angles, regardless of which side of the discontinuity they lie on. By employing this algorithm, the true rotational velocity can be accurately calculated, even when the Euler angles cross the discontinuity boundary.

2. **Velocity Signal Filtering**: The calculated angular velocity was then filtered using the selected fourth-order low-pass Butterworth filter with a cutoff frequency of 5 Hz. By filtering the velocity rather than the raw angle data, complications associated with the discontinuities in the angle signal are avoided.

3. **Angular Position Reconstruction** : The angular position signal was reconstructed through numerical integration of the filtered velocity signal.

4. **Centering** : Finally, the mean was subtracted from the reconstructed signal, to center it on the x-axis.

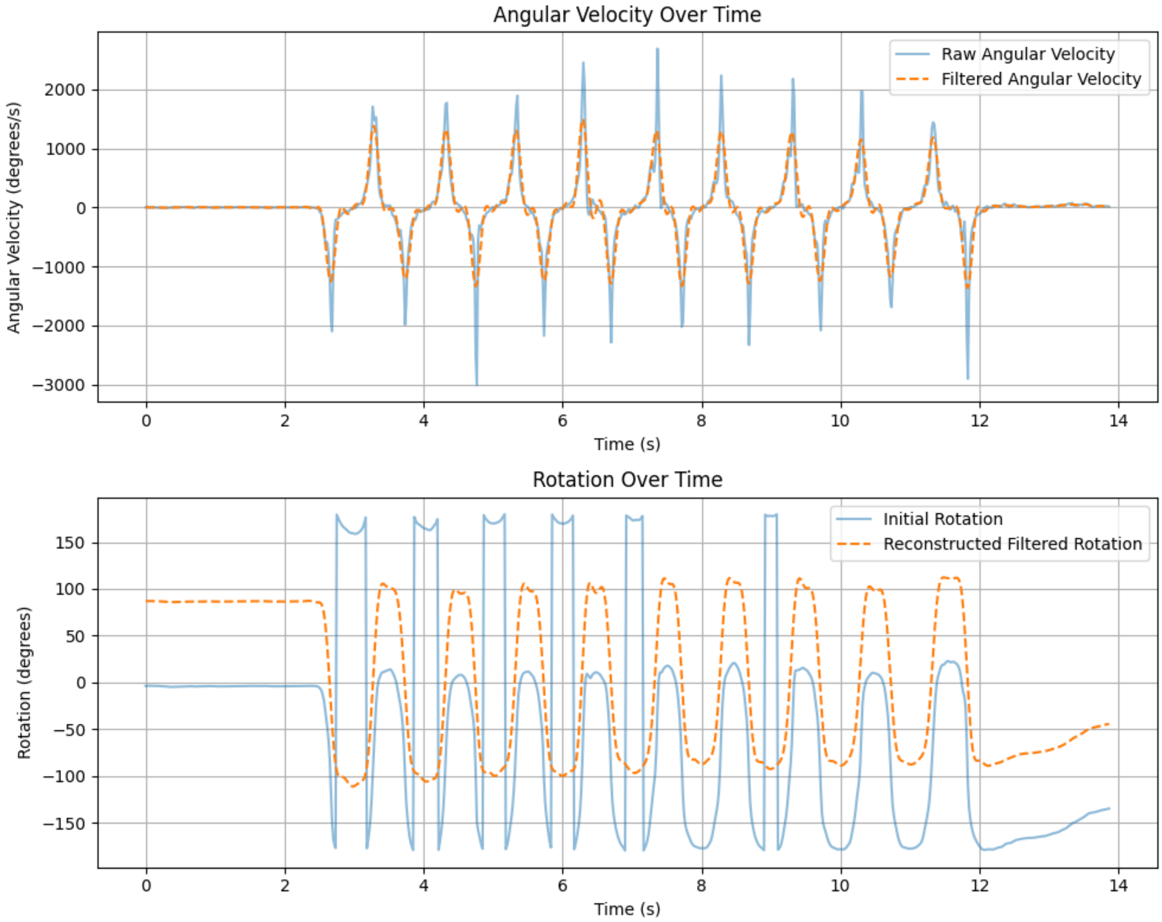


Figure 4.4: Comparison of raw and filtered signals for pronation-supination task. Top panel: Angular velocity over time, showing raw (solid line) and filtered (dashed line) velocities. Bottom panel: Rotation over time, displaying initial rotation (solid line) and reconstructed filtered rotation (dashed line). The filtered signals were obtained using a fourth-order Butterworth low-pass filter with a 5 Hz cutoff frequency. This approach addresses the challenges of Euler angle discontinuities by first calculating angular velocity, applying the filter, and then reconstructing the angular position through integration.

4.3.5.2 Spectral Analysis

Spectral analysis was performed on the movement data to quantify the frequency composition and Signal to Noise Ratio (SNR) of the recorded signals. The Fast Fourier Transform (FFT) [38] was applied to the preprocessed positional data for finger tapping and hand opening-closing tasks, and to the angular position data for pronation-supination task.

The power spectrum was calculated as the square of the magnitude of the FFT, normalized by the signal length. Signal power was quantified as the cumulative power within the frequency range of 0-5 Hz, which encompasses the primary movement frequencies observed in the tasks. Noise power was considered as the sum of power above 5 Hz. The SNR was calculated as the

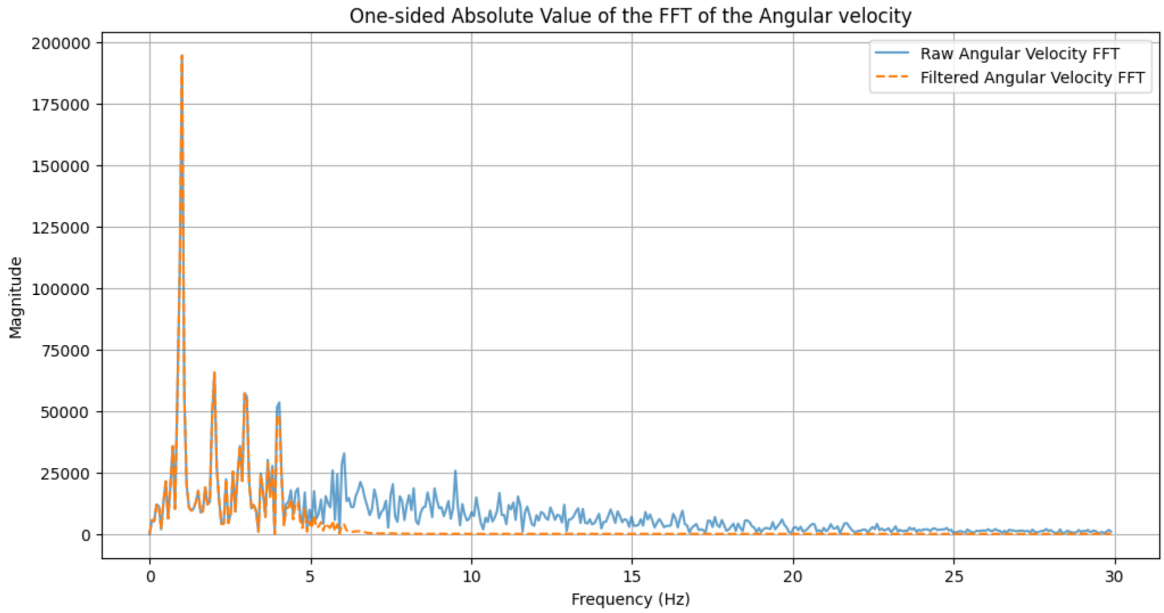


Figure 4.5: Comparison of FFT spectra for raw and filtered angular velocity signals. The plot shows the magnitude of the FFT against frequency for both the raw (solid line) and filtered (dashed line) angular velocity data.

ratio of signal power to noise power and expressed in decibels, providing a logarithmic scale that better represents the wide range of power ratios encountered in the data. The maximum frequency component within the 0-5 Hz range was designated as the characteristic movement frequency.

The application of the filtering process resulted in a minimum improvement of 10 dB in the signal-to-noise ratio across all tasks, as demonstrated in Figure 4.6. This substantial enhancement in SNR indicates a significant reduction in high-frequency noise, thereby improving the reliability of subsequent analyses.

4.4 Tracking Errors

A limitation of the Polhemus Patriot M is that only half of the total spatial sphere surrounding the source is practically usable at any given moment. Measurement ambiguities, typically manifesting as sign flips in the X, Y, Z measurements, occur when sensors traverse between hemispheres. This issue arises due to the symmetry of the magnetic fields generated by the source, resulting in two mathematical solutions to each set of sensor data processed. No ambiguity exists if the sensors operate solely within one hemisphere at a time. This parameter, termed the 'Hemisphere of Operation', can be set by the user.

The dataset in this study utilises the default Hemisphere of Operation, which is the positive

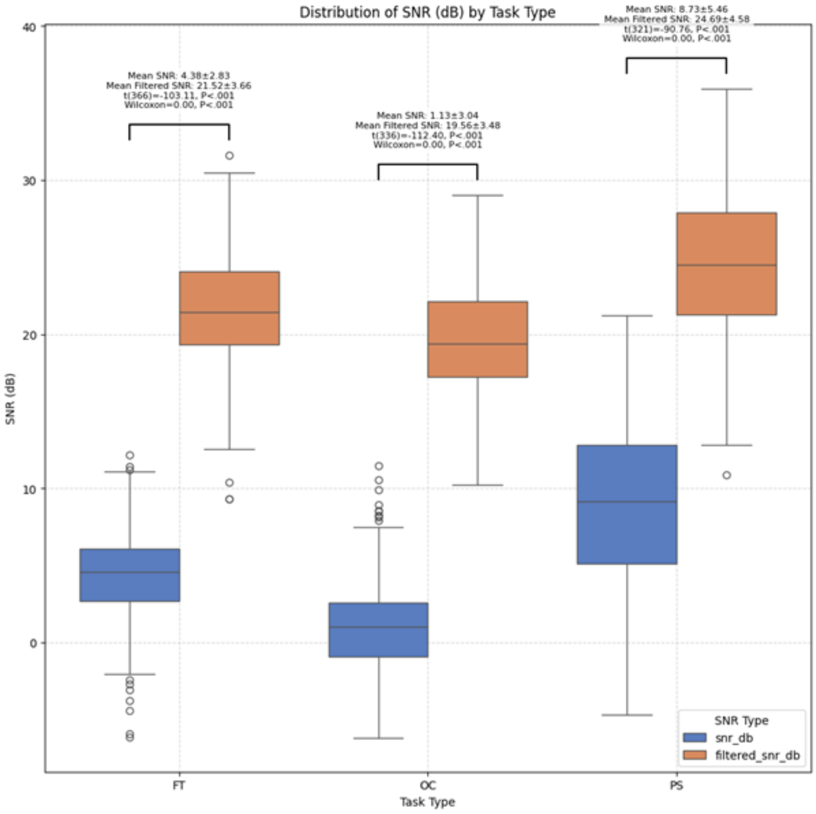


Figure 4.6: Signal to noise ratio for each task.

X or ”forward” hemisphere. In this configuration, the system does not report negative X measurements. Instead, when sensors are moved to the negative X side of the source, the Y and Z measurements are inverted.

Exploratory data analysis revealed that the experimental protocol did not sufficiently enforce this constraint. Some subjects were observed performing tasks in excessive proximity to the source, leading to boundary crossings and consequent measurement errors. This phenomenon was initially identified through unexpected fluctuations in the derived Euclidean distance.

The documentation highlights [140] that the ambiguity of the position usually results in the flipping of Y and Z coordinates. This was investigated with the following steps:

1. A function was defined that given a 1-D signal array, will return a Boolean mask the same length, in which a value is true if the current sign of the value has changed since the previous value, otherwise false.

2. For each sensor, two masks were generated for the y channel and z channel respectively. The logical AND operator was used to find samples for which both the y channel and z channel signs had flipped compared to the previous sample.

3. Any recordings for which this was true for any samples was marked as ‘potential hemi-

sphere switching'.

4. The distribution of maximum Euclidean distance vs. minimum x position for each recording was used to select and appropriate value to exclude recordings which showed this behaviour.

Figure 4.7 shows a pronation-supination recording, with samples annotated for each sensor that displayed YZ switching.The figure presents three subplots: from top to bottom, the Euclidean distance between the two sensors, the x position of sensor 1, and the x position of sensor 2.

The example in Figure 4.9 demonstrates that many of the points showing large spikes in Euclidean distance are accompanied by Y-Z inversions from one or both sensors. However, this is not universally the case. Therefore, this method is not indicative for all instances of larger than expected Euclidean distance. Additionally, the possibility exists that the trajectory of a sensor has moved it from the negative y and z quadrant to the positive y and z quadrant, not representing a case of hemisphere switching. Given these considerations, records that have any Y-Z inversion detections have been labeled as "potential hemisphere switching".

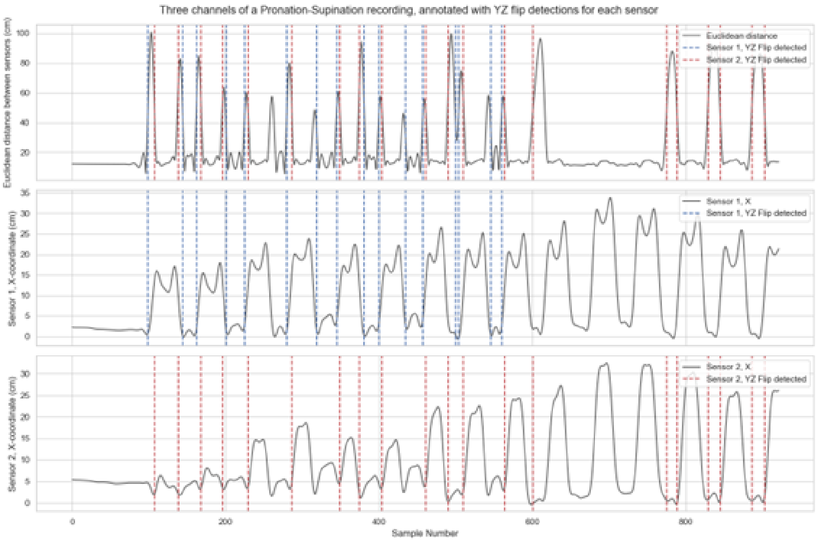


Figure 4.7: YZ flip detection for pronation supination movement.

Figure 4.9 illustrates the distribution of maximum Euclidean distance vs minimum x position for each recording. This graph is utilised to demonstrate that recordings with large spikes in the Euclidean distance tend to occur closer to the x-axis boundary. This was confirmed through the development of a visualisation application (Figure 4.8). Although not all recordings near the x-axis boundary exhibit these spikes, and some show spikes that are not necessarily near the x-axis boundary, it is possible that hemisphere crossing is not the only tracking issue within the dataset. Ferro-magnetic objects placed near the source can distort the magnetic field, which would also result in tracking issues.

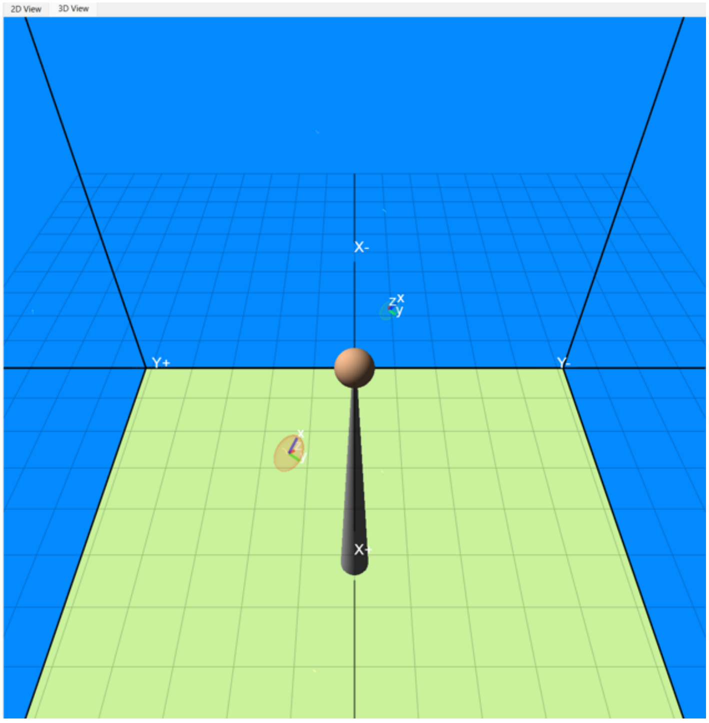


Figure 4.8: An application was developed in Python using the PyQt5 and PyQTGraph libraries to display real time and recorded Polhemus Patriot M data. The purpose of this application was to investigate artefacts found in a substantial amount of the recordings. Within the application window a 3D visualisation is presented. The orange sphere in the center of the window denotes the magnetic source, the camera position can be zoomed and panned around this point. Each sensor is represented by an ellipsoid, and translated and rotated according to its current pose. This image depicts the hemisphere switching issue.

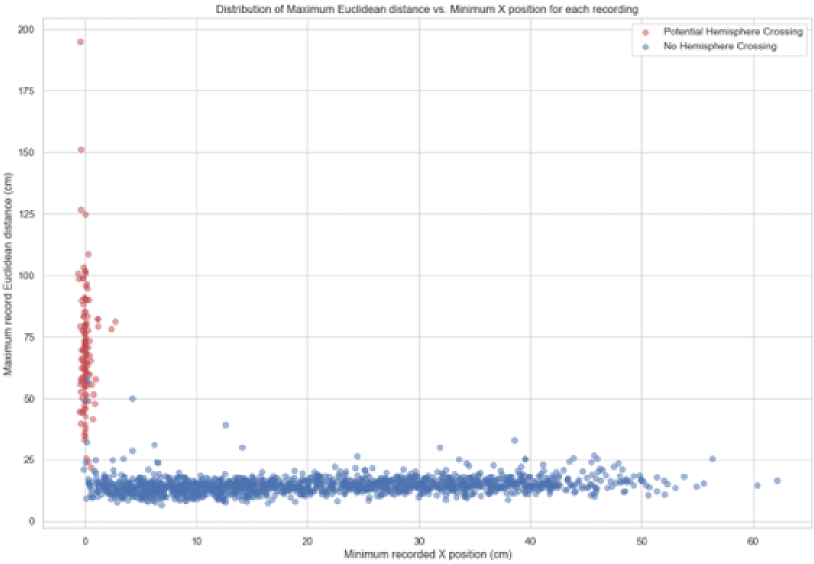


Figure 4.9: YZ flip distribution.

The investigation did not yield a wholly reliable and robust method for detecting these tracking anomalies. Various signal characteristics were examined, including the maximum velocity of each sensor, the maximum displacement (in a single timestep) of each sensor, and the minimum x position. However, these parameters did not provide distributions that offered a clear indication of the root cause of the issues. The most informative insight, which aligns with the description in the equipment manual, is that recordings in closer proximity to the x-boundary demonstrated a higher likelihood of sign flips and Euclidean distances exceeding plausible limits. Based on the distribution observed in Figure 4.9, an upper threshold of 25cm was established for the Euclidean distance between sensors. Consequently, any recordings exhibiting a Euclidean distance surpassing this threshold were excluded from further analysis to maintain data integrity. This has resulted in 110 (≈10%) recordings being removed from the dataset.

With these issues highlighted, greater focus should be given to the equipment setup. To the best of the author's knowledge, this is the first time the issue of hemisphere switching has been raised in similar studies using the Polhemus. It should be noted that the Patriot M also provides hemisphere tracking, a feature whereby Patriot M can continuously modify its operating hemisphere, given that it is started in a known valid hemisphere. It is advised that this mode be used in future studies. Additionally, it should also be noted that the magnetic source should be mounted in a fixed position to a non-metallic stand to minimise potential interference.

4.5 Data Preprocessing

Each recording comprises coordinate positions denoted as (x_{it}, y_{it}, z_{it}) captured over time steps labelled as $t = 1, 2, 3...N$ for sensors identified by i in the set $i \in \{1, 2\}$. These coordinates representing the distance in meters from the source, are first pre-processed before being used as inputs for the machine learning algorithms.

The preprocessing pipeline is designed to enhance the quality and consistency of the data set, comprising a sequence of steps aimed at noise reduction, outlier removal, trimming, data normalisation, positional calibration and windowing:

1. **Noise Reduction**: Each recording undergoes a smoothing process employing a 4th-order Butterworth filter with a cutoff frequency of 5Hz as outlined in Section 4.3.4
2. **Outlier Removal**: As outlined in Section 4.4 any recordings exceeding a separation distance of 0.25 m, due to assumed hemisphere switching. Figure 4.10 illustrates this issue again but with a finger-tapping recording.
3. **Trimming**: Transient behaviour is observed in the beginning of most recordings. This is removed by setting a velocity threshold, that once exceeded by either sensor de-

notes the beginning of the recording. This process is illustrated in Fig 4.11, which demonstrates the trimming of a finger tapping task. The velocity of each sensor is derived by first calculating the euclidean distance between each sensor and the source $d_{it} = \sqrt{x_{it}^2 + y_{it}^2 + z_{it}^2}$, followed by the first order derivative $v_{it} = \frac{d_{it} - d_{i(t-1)}}{\Delta T}$, where ΔT represents the time difference between two samples (for a sampling rate of 60 Hz this is $16.667ms$).

4. **Data Normalisation**: To normalise these coordinates, they are scaled to fall within the range of $[-1, 1]$ by multiplying each axis by 0.5, effectively mapping each axis of the coordinate positions into a cube with side length of 2 meters centered around the origin. This scaling is sufficient for encompassing the tracking range of the Polhemus Patriot M.

5. **Positional Calibration**: Initially, the raw data is oriented with respect to a magnetic source located 0.8 meters in front of the subject. To mitigate potential positional bias, this reference point is redefined to be the initial position of sensor 1 ($j = 0$ in the trimmed recording). Whereby all data points are translated based on this new reference position.

6. **Windowing**: The machine learning approaches used require the input data to be of a fixed length. In this aspect, the recordings in the dataset varied in length considerably. Ranging from 2 seconds to 35 seconds, with a mean of 12.2 seconds and a standard deviation of 4.6 seconds. To retain as many of the recordings as possible, the window length was set to 5 seconds, resulting in a further 28 recordings removed from the dataset that were shorter than this threshold.

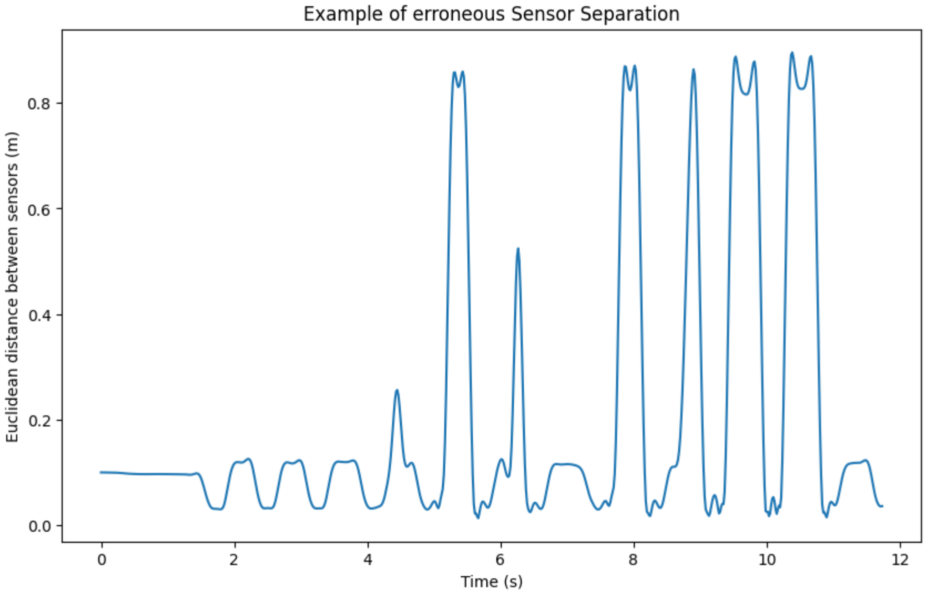


Figure 4.10: This figure presents an example of erroneous positional data. The separation between the sensors exceeds the separation threshold of 0.25 m, indicating that the sensor has been identified in the incorrect hemisphere of the Polhemus Patriot M.

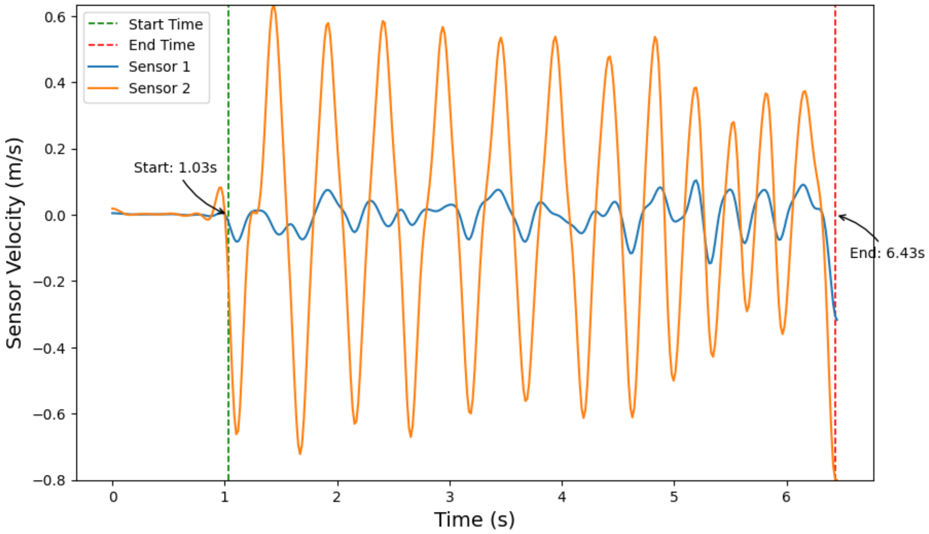


Figure 4.11: The figure displays a plot of the positional velocities from both sensors during a finger tapping recording. A threshold velocity of 0.2 m/s, was used to remove transient behaviour not associated with the task from the data.

4.6 Dataset Generation

4.6.1 Input Timeseries

Following data preprocessing, each sample was transformed into a 2D array with the shape $(6, 300)$. This shape encapsulates information from two sensors, each providing three channels of spatial coordinates (x, y, z). These measurements were recorded over a span of 5 seconds, sampled at a rate of 60 Hz, yielding 300 data points per channel. Figure 4.12 exemplifies the data input for a pronation-supination task, illustrating the variations in MDS-UPDRS severities.

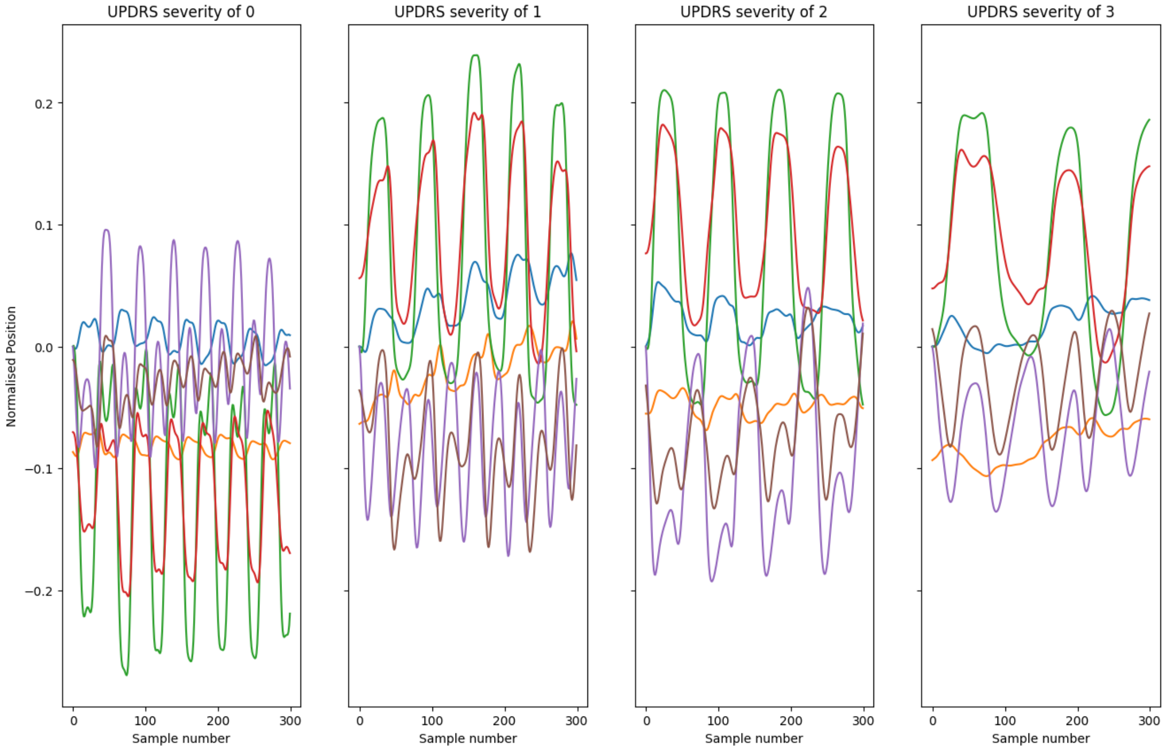


Figure 4.12: Data input example for a pronation-supination task, and varying MDS-UPDRS severities.

4.6.2 Data Augmentation

In the process of preparing the data for model training it was observed that the distribution of data for the bradykinesia targets was sufficiently balanced across task types (Table 4.3). However, for distinguishing between PD and NC the dataset showed a considerable imbalance between different task types (as expected due to the larger number of PD subjects). This necessitated a strategy to improve the parity between the two groups to ensure a more reliable and generalised training process.

Task Type	Bradykinesia	Count
Pronation-supination	Present	180
	Absent	150
Hand opening and close	Present	180
	Absent	177
Finger tapping	Present	194
	Absent	163

Table 4.3: Distribution of data representing bradykinesia classes across different task types.

To address this, a windowing technique was used on control recordings, wherein each recording was segmented into a series of overlapping windows with 60% overlap. Additionally, this was also only applied to healthy controls, as the effects of fatigue wasn't expected to impact them. This was limited to four windows per recording to prevent over-representation of any one subject. Table 4.4, shows the notable improvement in class balance due to this approach.

Task Type	Diagnosis	Count	Augmented Count
Pronation-supination	PD	271	271
	NC	59	202
Hand opening and close	PD	281	281
	NC	76	247
Finger tapping	PD	279	279
	NC	78	259

Table 4.4: Distribution of PD and NC groups for different task types before and after data augmentation.

4.7 Model Implementation and Training Strategy

4.7.1 Experimental Setup

The study utilised Python 3.9.13 and the tsai library [129] (version 0.3.7) for model implementations. The tsai library was chosen for its specialised focus on time series analysis and its integration with PyTorch (version 2.0.0, cuda 11.7) and fastai API (version 2.7.12), offering a robust foundation for implementing state-of-the-art time series classification models. These were the 'Plus' implementations of MiniRocket, MutliRocket and InceptionTime. Experiments were conducted on a system featuring an AMD Ryzen 9 3900 CPU, an Nvidia 3080 GPU with 10GB VRAM, and 32GB of RAM.

4.7.2 Cross-Validation Strategy

To assess model performance and generalisation, a nested cross-validation procedure was implemented. Nested cross-validation was chosen to provide a more robust estimate of model performance and to avoid overfitting during hyperparameter tuning [104]. The dataset was partitioned into 5 outer folds and 3 inner folds, striking a balance between computational efficiency and robust validation. This nested structure is crucial as it prevents information leakage between outer folds during hyperparameter tuning, thus avoiding biased performance estimates.

The StratifiedGroupKFold function from the scikit-learn library [137] was employed for partitioning. This approach ensures each subject appears exactly once in the test set across folds while attempting to preserve the original dataset's sample distribution. To maintain consistency, identical splits were generated for each target and task across all models.

4.7.3 Model Parameters and Hyperparameter Optimisation

Model parameters are outlined in Table 4.5. For hyperparameter optimisation, Bayesian optimisation was implemented using the Optuna library [2]. Bayesian optimisation was employed for its efficiency in exploring complex hyperparameter spaces, allowing for a more intelligent search than grid or random search methods. A total of 50 trials were executed to explore the hyperparameter space systematically.

In each trial, a unique combination of hyperparameters was sampled from a predefined search space (Table 4.6). These hyperparameters were then evaluated within the nested cross-validation framework. For each inner fold, a loss metric was computed, and these losses were averaged to produce a single scalar value representing the overall performance for the given hyperparameters.

Initially, internal model parameters in Table 4.5 were included in this search. However, as results were significantly poorer compared to default values (perhaps the number of trials was not enough given the search space), they were left unchanged for this experiment and reserved for future investigation.

4.7.4 Training Process

The training process leveraged the one-cycle learning rate scheduler proposed by Smith [173], implemented as the fit_once_cycle method. The one-cycle scheduler was chosen for its ability to achieve faster convergence and better generalisation. Categorical Cross-Entropy was employed as the loss function. To address class imbalance, weights were adjusted using inverse

Parameter	MiniRocket	MultiRocket	InceptionTime
Number of features/filters	10,000	50,000	32
Batch normalisation	True	True	True
Custom head	None	None	None
Dropout rate for fully connected layer	0.0	0.0	0.0
Max dilations per kernel	32	32	-
Kernel size	9	9	-
Maximum number of channels	None	None	-
Maximum number of kernels	84	84	-
Same number of features per kernel stride	False	False	-
Long Short-term Attention Z-Pool	False	False	-
Zero weight initialisation	True	True	-
Calculate first order derivative	-	True	-
Output Flattened	-	-	False
Concatenated Pooling	-	-	False
Output Value Range	-	-	None
Depth	-	-	6

Table 4.5: The default model parameters for MiniRocket, MultiRocket, and InceptionTime.

Hyperparameter	Search Space
Learning Rate	$[10^{-5}, 10^{-1}]$ (log space)
Number of Epochs	$\{16, 32, 64\}$
Batch Size	$\{10, 11, 12, 13, 14, 15\}$

Table 4.6: Hyperparameter search space

frequency weights.

The Adam optimizer [88] was used with a learning rate of 0.001, $\beta_1 = 0.9$, $\beta_2 = 0.99$, epsilon $= 1 \times 10^{-5}$, and weight decay = 0.01. These are the default parameters in fastai and were chosen based on their proven effectiveness in similar deep learning applications [62].

4.8 Statistical Analysis

For each test, a p-value less than or equal to the significance level ($\alpha \leq 0.05$) is considered statistically significant.

4.8.1 Model Performance

A critical difference diagram (CCD) [47], commonly used in timeseries classification literature for comparing model performance. will be generated for each target and task. The nested splits generated are identical. In this approach, each outer fold is treated as an observation, and accuracy is compared between each model. CCD's in two steps. First the Friedman test (non-parametric one way repeated analysis of variance by ranks) is computed, which indicates whether there is any significant difference between the models. If the test rejects this hypothesis, then the Wilcoxon signed rank test is used to determine whether each pair of models exhibits a significant difference. Given that multiple pariwise compairsons are made, the Wilcoxon test is adjusted using Holm's method.

One limitation of this approach is that, treating each fold as a separate dataset violates the assumption of independence of observations, as the test data used is included in four folds for training. This violation increases the probability of Type I errors [49]. Thus, caution must me given when interpreting these results.

4.8.2 Misclassification Analysis

Investigating the factors influencing false positives and negatives is vital in ensuring that the trained models exhibit expected behaviour, and in understanding their potential limitations.

A critical variable in this context is the UPDRS severity score. It is anticipated that an increase in the UPDRS severity score would consequently decrease the occurrence of false negatives for both targets. This hypothesis is grounded in the understanding that higher severity scores, indicative of pronounced bradykinesia symptoms, should facilitate a more accurate classification. For each target, the following subgroups are identified, in which this association will be tested:

- **Diagnosis prediction**: The subgroup encompasses PD samples, in which true positives are samples being correctly predicted as PD, and false negatives are instances where a sample is misclassified as healthy control.
- **Bradykinesia prediction**: The subgroup comprises samples with a UPDRS score greater than zero, in which true positives are samples being correctly predicted as having a score greater than zero, and false negatives are instances where a sample is misclassified a score equal to zero.

The Cochran-Armitage test is suitable for establishing the significance of this relation, whereby associations between a binary variable (true positives, false negatives) and an ordinal variable (UPDRS severity score, having five distinct levels) [8].

Additionally, for experiments where UPDRS severity is the target, the distribution of diagnoses in false positives (predictions that surpass the score of zero for samples that were assigned a score of zero) was examined. This analysis aimed to determine whether a significant discrepancy exists between misclassifications of healthy controls and PD samples. Such an investigation could potentially reveal the model's capability to detect early, subtle signs of PD that might lead to misclassifications. Pearson's chi-squared test was employed to establish whether the number of misclassifications between PD and HC groups were independent.

Given that distinct subgroups are identified for statistical tests regarding experiments in which UPDRS severity is the target, the multiple comparisons problem is not applicable, negating the necessity for corrections.

4.9 Results

Results are reported as mean ± standard deviation, computed across the 5 outer folds of the nested cross-validation. This approach provides a robust estimate of model performance while accounting for variability across different data partitions. Table 4.8 presents the results for each task and model when the target is diagnosis, while Table 4.10 shows the results when the target is bradykinesia (present vs. absent).

To visualise the effect of MDS-UPDRS severity on model performance, Figure 4.14 illustrates the relationship between severity and false negatives for each task and model when the diagnosis is PD. Similarly, Figure 4.17 demonstrates the effect of MDS-UPDRS severity on misclassifications for each task and model from the severity prediction task.

ROC curves are plotted for each target, with Figures 4.13 and 4.16 representing diagnosis and bradykinesia, respectively.

4.9.1 PD vs. Healthy Controls

To provide context for the subsequent analyses, the distribution of UPDRS severity levels across different task types and diagnoses is presented in Table 4.7.

Task Type	Diagnosis	UPDRS Severity				
		0.0	1.0	2.0	3.0	4.0
Hand opening and closing	NC	245	2	-	-	-
	PD	102	97	60	20	2
Pronation-supination	NC	198	4	-	-	-
	PD	92	90	59	28	2
Finger tapping	NC	259	-	-	-	-
	PD	85	97	71	26	-

Table 4.7: Distribution of UPDRS severity levels across different task types and diagnosis for the **augmented** dataset.

Model	Accuracy	Precision	Recall	F1 score	MCC	AUC (ROC)
Pronation-Supination						
InceptionTime	**0.84 ± 0.05**	**0.86 ± 0.11**	0.88 ± 0.06	**0.87 ± 0.04**	**0.69 ± 0.10**	**0.93 ± 0.04**
MiniRocket	0.80 ± 0.05	0.82 ± 0.10	0.87 ± 0.09	0.83 ± 0.05	0.62 ± 0.09	0.90 ± 0.04
MultiRocket	0.82 ± 0.06	0.82 ± 0.12	**0.90 ± 0.06**	0.85 ± 0.05	0.65 ± 0.11	0.93 ± 0.03
Hand Opening and Closing						
InceptionTime	0.77 ± 0.05	0.75 ± 0.10	0.87 ± 0.05	0.80 ± 0.06	0.55 ± 0.07	0.86 ± 0.03
MiniRocket	0.79 ± 0.05	0.77 ± 0.09	0.86 ± 0.04	0.81 ± 0.05	0.58 ± 0.10	0.87 ± 0.04
MultiRocket	**0.81 ± 0.04**	**0.79 ± 0.08**	**0.88 ± 0.06**	**0.83 ± 0.05**	**0.63 ± 0.08**	**0.89 ± 0.03**
Finger Tapping						
InceptionTime	0.75 ± 0.07	0.73 ± 0.11	0.85 ± 0.07	0.78 ± 0.06	0.49 ± 0.15	0.80 ± 0.10
MiniRocket	0.78 ± 0.07	0.75 ± 0.11	0.84 ± 0.11	0.79 ± 0.09	0.54 ± 0.11	0.85 ± 0.06
MultiRocket	**0.81 ± 0.08**	**0.77 ± 0.11**	**0.89 ± 0.06**	**0.83 ± 0.08**	**0.61 ± 0.13**	**0.87 ± 0.06**

Table 4.8: Comparative performance of InceptionTime, MiniRocket and MultiRocket in distinguishing between PD and healthy controls on three distinct motor tasks. The results represent the mean values derived from 5 outer folds of a nested cross-validation method, with the standard deviation also presented. Metrics encompass accuracy, precision, recall, F1 score, Matthews Correlation Coefficient (MCC) and Area Under the Reciever Operating Characteristic Curve (AUC-ROC). Bold values indicate the best performance across models for each task in the respective metric column.

In the pronation-supination task, the InceptionTime model demonstrated superior performance across most metrics, with MultiRocket slightly outperforming it in recall. To assess whether these differences were statistically significant, the Friedman test was employed. This non-parametric test was chosen because it does not assume normality of the data and is suitable for comparing multiple related samples. The test did not reject the null hypothesis ($p = 0.07$), indicating that despite the observed differences, there was no statistically significant variation in model performance. This result suggests that all three models perform comparably well for this task.

Similar patterns were observed for the hand opening and closing task, where MultiRocket led in all metrics, and the finger tapping task, where MultiRocket again dominated. In both cases, the Friedman test failed to find significant differences ($p = 0.25$ and $p = 0.33$, respectively). These consistent findings across tasks suggest that while there are observable differences in performance metrics, all three models are similarly capable in differentiating PD from healthy controls across various upper limb tasks.

To investigate the relationship between disease severity and model performance, the Cochran-Armitage test for trend was employed. This test was chosen because it can detect a linear trend in binomial proportions across ordinal categories, making it ideal for examining how false negative rates change with increasing UPDRS severity. Significant ordinal associations were found for all models across all tasks, with p-values ranging from < 0.001 to 0.05. These results suggest that as UPDRS severity increases, the models become more accurate in identifying PD cases. This finding aligns with clinical expectations, as more severe cases typically exhibit more pronounced symptoms that are easier to detect.

Figure 4.13 showcases the mean ROC curves alongside the variability observed across the 5 outer folds of the nested cross-validation method for each task and model. The gray shaded area enveloping the mean ROC curve represents the standard deviation of the true positive rate, providing insight into the stability of the models' performance across different data partitions.

Figure 4.14 illustrates the distribution of false negatives for the positive case (PD), grouped by UPDRS severity levels for each task. This visualisation aids in understanding how the ordinal severity scores affect the predictive accuracy of each model, further supporting the findings from the Cochran-Armitage test.

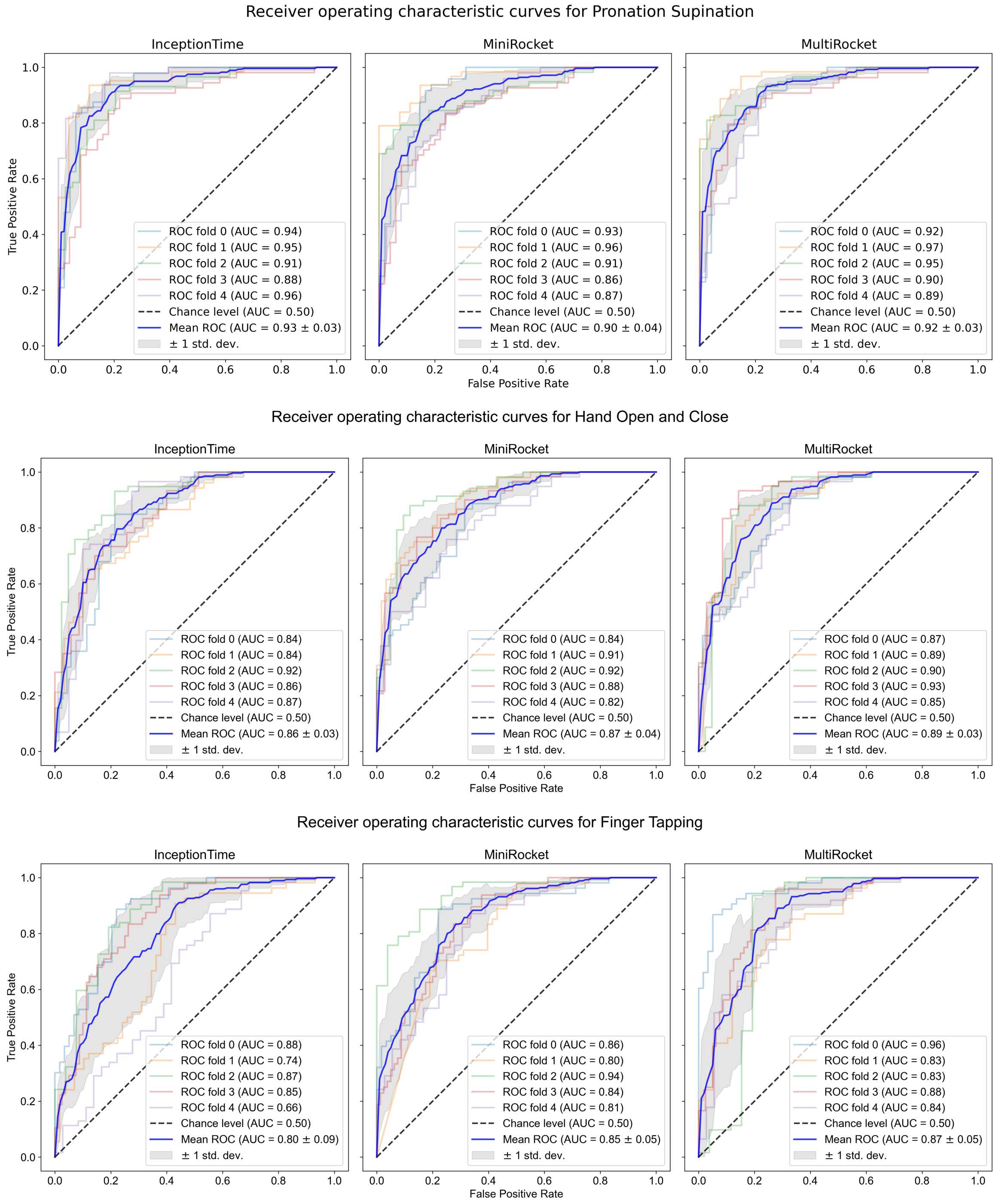


Figure 4.13: The plot showcases the mean ROC curves alongside variability observed across 5 outer folds of a nested cross-validation method, for each task and model. Each model is trained to distinguish between healthy controls and PD for each given task. The gray shaded area enveloping the mean ROC curve represents the standard deviation of the true positive rate. The chance level is distinctly marked to facilitate a comparison against a model operating at a level equivalent to random guessing. For context, an AUC 0.70 to 0.80 are 'acceptable', 0.80 to 0.90 'excellent' and 0.9 or above 'outstanding' [97].

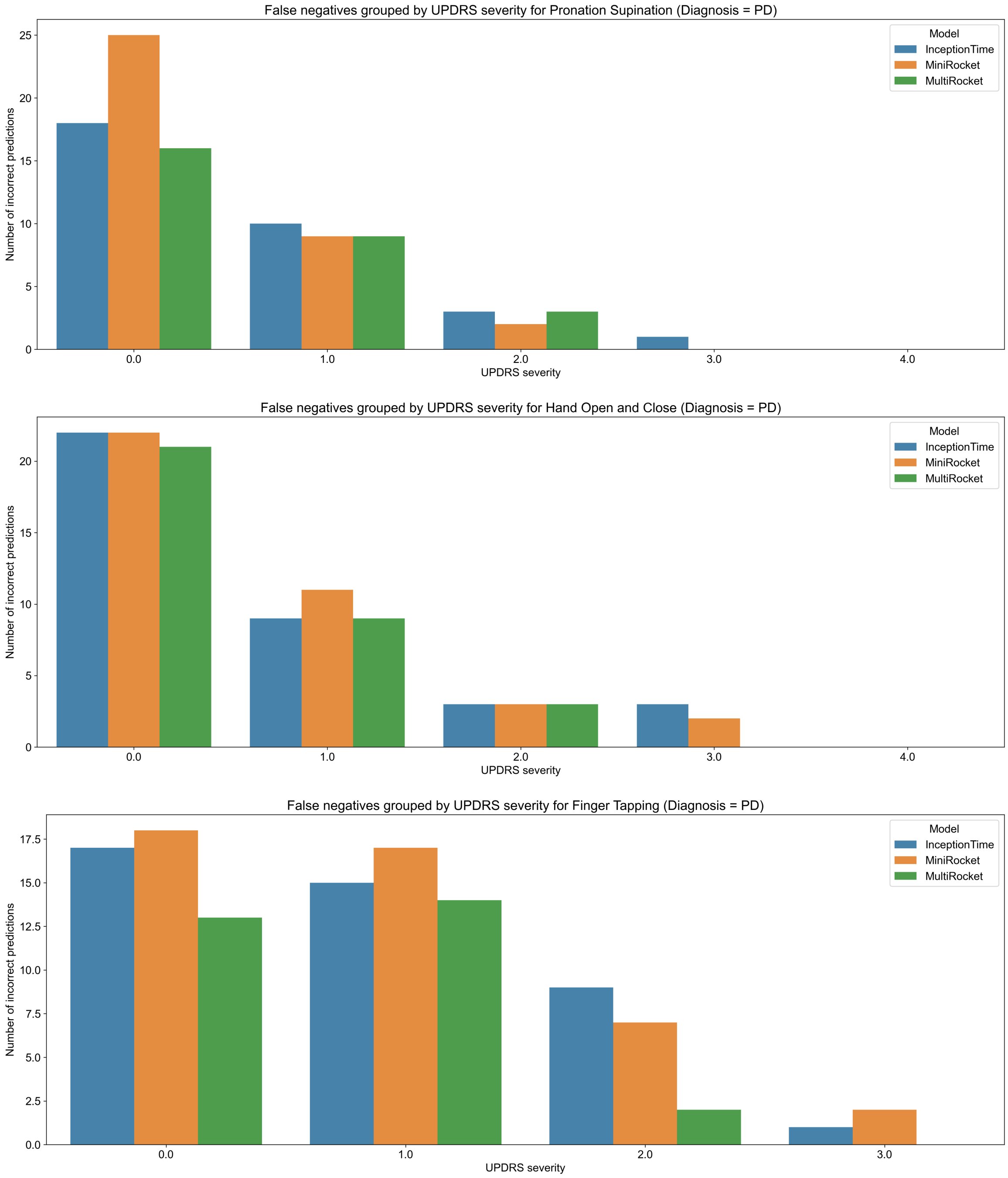


Figure 4.14: Bar plots illustrating The number of false negatives for the positive case (PD), grouped by UPDRS severity levels for each task. This visualisation aids in understanding how the ordinal severity scores effects the predictive accuracy of each model (denoted by different coloured bars). It is important to note that the data represents an aggregation of all false positives accumulated from each fold of the nested cross-validation. Additionally the finger tapping dataset lacked samples with severity level equal to 4.

4.9.2 Identifying Clinically Slight Bradykinesia

In this analysis, the focus shifted to a more challenging task: distinguishing between normal (UPDRS score of 0) and any level of bradykinesia (UPDRS score > 0). This binary classification is particularly relevant for early detection and monitoring of PD progression.

Table 4.9 presents the distribution of UPDRS severity levels across different task types and diagnoses for the original dataset, providing context for the subsequent analyses of slight bradykinesia detection.

Table 4.10 shows the comparative performance of InceptionTime, MiniRocket, and MultiRocket in distinguishing between severity equal to 0 or greater than 0 for the three distinct motor tasks. The results are presented using the same set of performance metrics as in the previous analysis.

Task Type	Diagnosis	UPDRS Severity				
		0.0	1.0	2.0	3.0	4.0
Hand opening and closing	NC	75	1	-	-	-
	PD	102	97	60	20	2
Pronation-supination	NC	58	1	-	-	-
	PD	92	90	59	28	2
Finger tapping	NC	78	-	-	-	-
	PD	85	97	71	26	-

Table 4.9: Distribution of UPDRS severity levels across different task types and diagnosis for **original** dataset.

Model	Accuracy	Precision	Recall	F1 score	MCC	AUC (ROC)
Pronation-Supination						
InceptionTime	0.80 ± 0.05	0.82 ± 0.05	0.81 ± 0.06	0.81 ± 0.04	0.59 ± 0.11	0.89 ± 0.05
MiniRocket	**0.82 ± 0.06**	**0.87 ± 0.09**	0.80 ± 0.04	**0.83 ± 0.05**	**0.65 ± 0.13**	**0.90 ± 0.07**
MultiRocket	0.82 ± 0.06	0.84 ± 0.07	**0.83 ± 0.06**	0.83 ± 0.06	0.63 ± 0.12	0.90 ± 0.05
Hand Opening and Closing						
InceptionTime	0.74 ± 0.05	0.75 ± 0.12	0.75 ± 0.06	0.74 ± 0.04	0.49 ± 0.10	0.83 ± 0.06
MiniRocket	0.79 ± 0.05	0.78 ± 0.11	**0.82 ± 0.08**	0.79 ± 0.05	0.58 ± 0.10	0.86 ± 0.04
MultiRocket	**0.80 ± 0.05**	**0.81 ± 0.11**	0.80 ± 0.07	**0.80 ± 0.06**	**0.61 ± 0.11**	**0.87 ± 0.04**
Finger Tapping						
InceptionTime	**0.71 ± 0.06**	0.74 ± 0.15	**0.74 ± 0.07**	**0.73 ± 0.07**	**0.43 ± 0.11**	0.77 ± 0.06
MiniRocket	0.70 ± 0.02	**0.75 ± 0.11**	0.68 ± 0.07	0.71 ± 0.04	0.42 ± 0.04	0.79 ± 0.02
MultiRocket	0.70 ± 0.05	0.74 ± 0.12	0.69 ± 0.10	0.71 ± 0.07	0.41 ± 0.10	**0.81 ± 0.02**

Table 4.10: Comparative performance of InceptionTime, MiniRocket and MultiRocket in distinguishing between severity equal to 0 or greater than 0 for three distinct motor tasks. The results represent the mean values derived from 5 outer folds of a cross-validation method, with the standard deviation also presented. Metrics encompass accuracy, precision, recall, F1 score, Matthews Correlation Coefficient (MCC) and Area Under the Reciever Operating Characteristic Curve (AUC-ROC). Bold values indicate the best performance across models for each task in the respective metric column.

For the pronation-supination task, MiniRocket exhibited the best average performance for most metrics, but was surpassed by MultiRocket for recall. The drop in AUC (ROC) scores for fold 1 as illustrated in Fig 4.16 is particularly notable in comparison to the other folds, suggesting that recordings from other subjects do not generalise as well to the ones in the test set. Overall, the performance between all models for this task are very similar. Expectedly, the Friedman test did not find a significant difference in the performance of the different models (p = 0.82).

For the hand opening and closing task, MultiRocket had the best average performance for all metrics except for recall. Here, the Friedman did find a significant difference in model accuracies (p=0.02). To further investigate these differences, pairwise Wilcoxon signed-rank tests with Holm correction were conducted. These tests were chosen for their ability to compare paired samples without assuming normality, with the Holm correction accounting for multiple comparisons. Surprisingly, despite the significant Friedman test result, no pairwise comparisons reached statistical significance. This is visually represented in Fig 4.15, which displays a critical difference diagram summarising the statistical analysis.

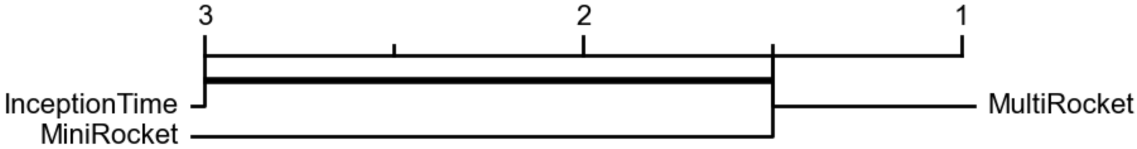


Figure 4.15: The critical difference diagram depicts a comparative analysis of model accuracy in bradykinesia detection for the hand opening and closing task. The Friedman test was used to rank the models across all folds. MultiRocket and MiniRocket achieved a tied average rank of 1.5, surpassing InceptionTime with a rank of 3. Subsequent post-hoc analysis with pairwise Wilcox signed-rank tests with Holm correction, reveled no statistical distance between any models. Delineated by the line connecting all models.

For the finger tapping task, InceptionTime showed the best performance over most tasks, except for recall and ROC AUC. Again, the Friedman test did not find a significant difference in the performance of the different models (p=0.33).

The Cochran-Armitage test was again employed to examine the relationship between UPDRS severity and false negatives, revealing significant ordinal associations for all models and tasks (all reported, $p < 0.001$). This consistent finding across both classification tasks (PD vs. Healthy Controls and identifying slight bradykinesia) underscores the strong relationship between low severity scores and worse model accuracy.

Finally, to explore potential biases in misclassification, Pearson's chi-squared tests were used to examine the distribution of diagnoses in false positives (when a subject performed the task with no abnormalities, but were classed as abnormal). This test was chosen for its ability to assess independence between categorical variables. The results varied across tasks and models, with one showing a significant difference MultiRocket in finger tapping, ($p = 0.002$) and others showing no significant difference. These findings don't majorly significantly support the idea that misclassification patterns may differ between PD and healthy control groups due to subtle, early signs of PD that the models might be detecting.

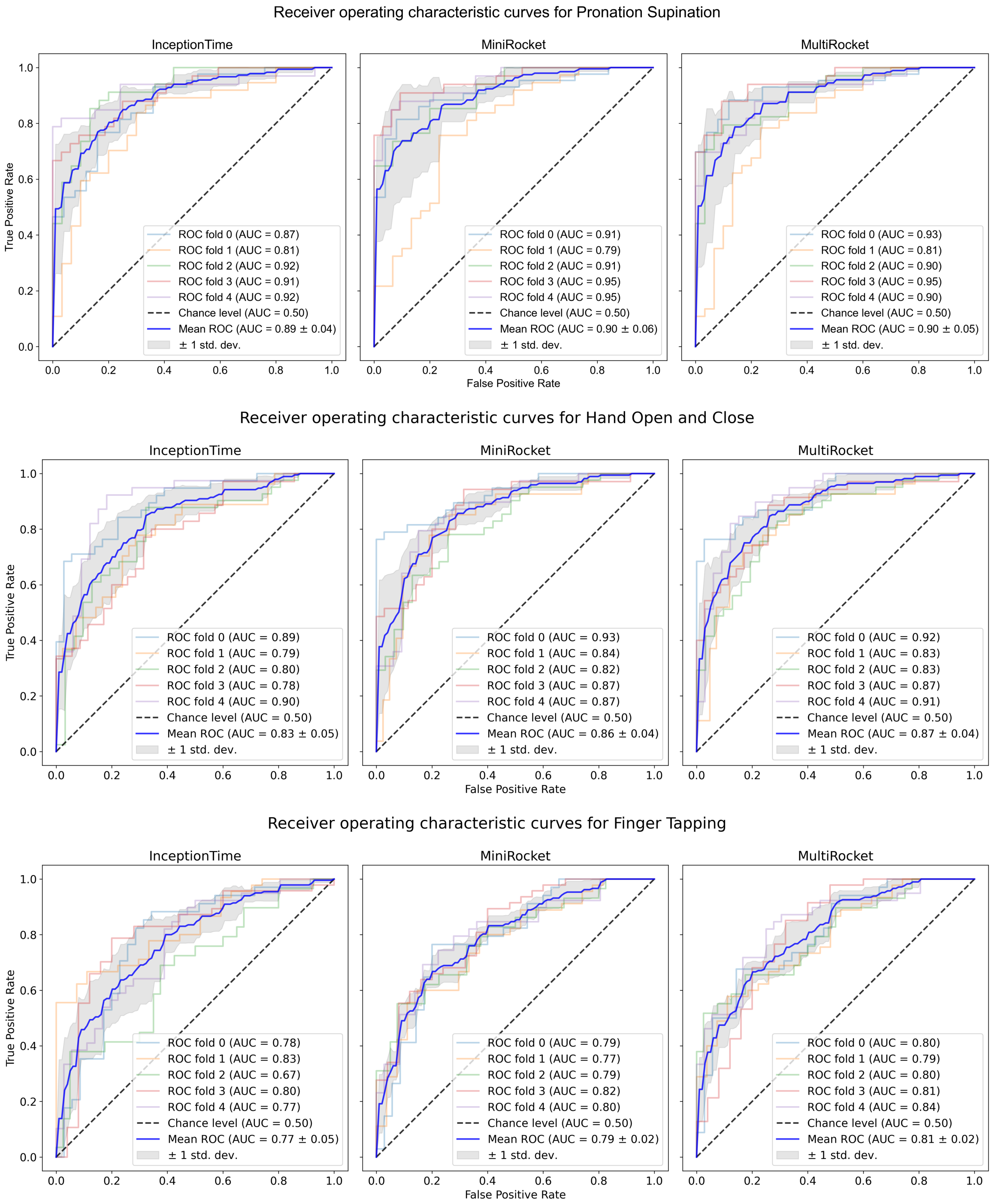


Figure 4.16: The plot showcases the mean ROC curves alongside variability observed across 5 outer folds of a nested cross-validation method, for each task and model. Each model is trained to distinguish between healthy controls and PD for each given task. The gray shaded area enveloping the mean ROC curve represents the standard deviation of the true positive rate. The chance level is distinctly marked to facilitate a comparison against a model operating at a level equivalent to random guessing. For context, an AUC 0.70 to 0.80 are 'acceptable', 0.80 to 0.90 'excellent' and 0.9 or above 'outstanding [97].

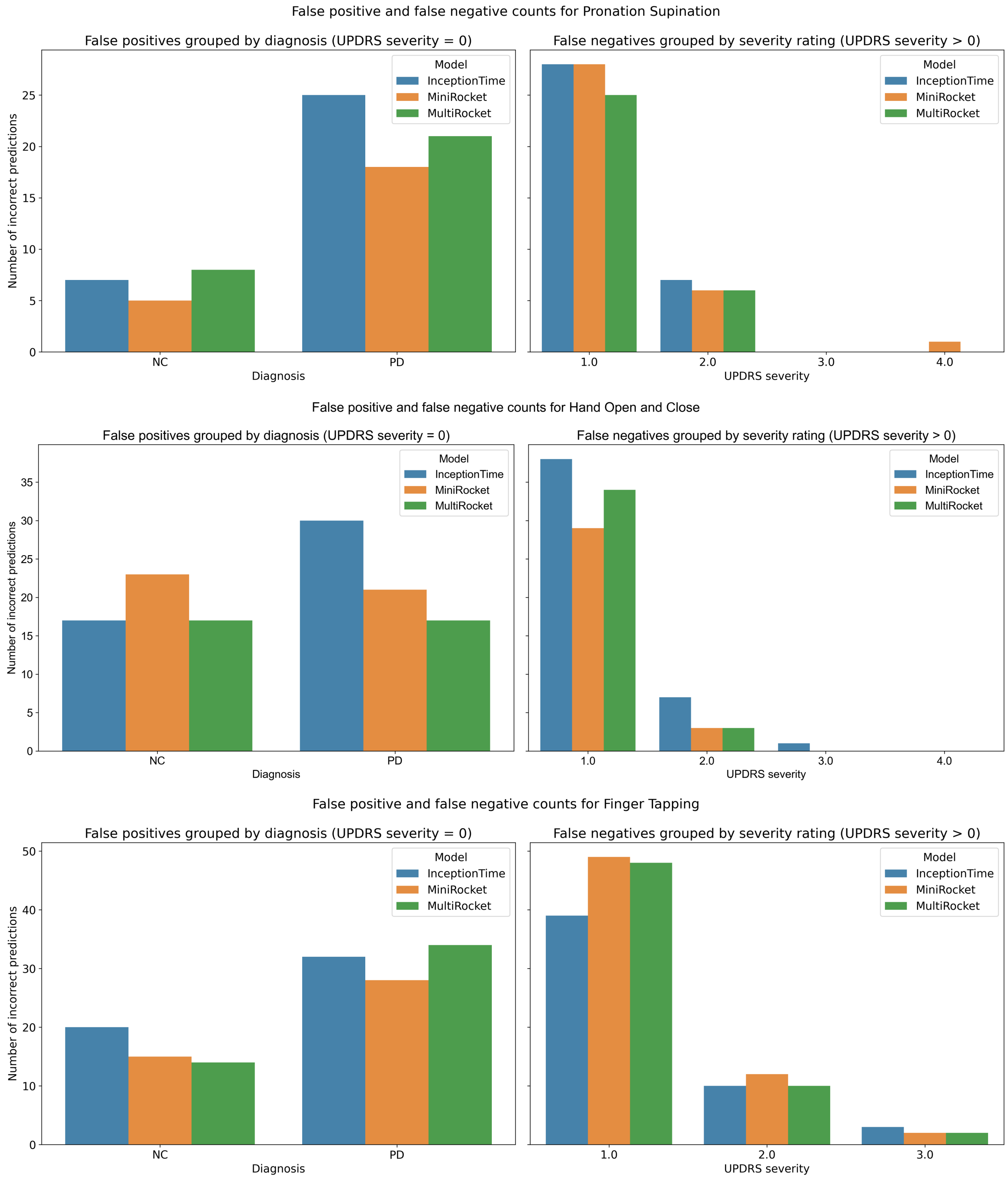


Figure 4.17: Dual bar plots delineating the distribution of false negatives and false positives predictions. The plot on the left shows false negatives (UPDRS greater than 0) grouped by diagnosis. The plot on the right shows false positives (UPDRS equal to 0). This visualisation aids in understanding how diagnosis and ordinal severity scores effect the predictive accuracy of each model (denoted by different coloured bars). It is important to note that the data represents an aggregation of all false positives accumulated from each fold of the nested cross-validation. Additionally the finger tapping dataset lacked samples with severity level equal to 4.

4.10 Discussion

4.10.1 Model Performance

This study evaluates the effectiveness of InceptionTime, MiniRocket and MultiRocket in categorising two key clinical parameters pertinent to movement recordings: diagnostic status as either PD or HC, and the severity of updrs scores. In this regard the presence or absence of bradykinesia, as defined by the MDS-UPDRS scoring criteria (bradykinesia is considered to range from slight to severe). The methods were evaluated based on five criteria: (1) comparative performance between tasks; (2) performance across three distinct tasks; (3) the correlation between severity scores and misclassification of PD; (4) the distribution of diagnoses in instances of misclassified no bradykinesia recordings; and (5) the relationship between MDS-UPDRS severity scores and instances where bradykinesia was misclassified as present.

Statistical analysis revealed no significant differences between InceptionTime, MiniRocket, and MultiRocket models at both the target and task level. This lack of statistically significant variation in performance suggests that all three models demonstrate comparable efficacy in analysing upper limb movement data for PD diagnosis and the detection of slight bradykinesia. The lack of statistical difference may be due in part in how fundamentally each model seeks to generate convolutional kernels that extract discriminatory features within the time-series data, albeit varying in their approach. InceptionTime, as a deep learning method, can learn and adapt its kernels during training. In contrast MiniRocket, on the other hand, employs a predefined set of kernels, while MultiRocket expands on this with additional pooling operations and the computation of first-order derivatives from the input time series, resulting in five times the number of features.

The performance of models was largely comparable across task types and model architectures, with one notable exception. The identification of bradykinesia in finger tapping recordings, showed a significant decline in accuracy and MCC metrics compared to other tasks. This is evident in the misclassification analysis, where a larger proportion of recordings with a severity score of 1 were incorrectly classified as having a severity score of 0. This observation may be linked to challenges encountered in developing an approach for segmenting the 1-D representation of the finger-tapping signal. During this phase, it was noted that it was particularly difficult to visually distinguish between severity scores for finger tapping. It appeared that subject tended to prioritise the "as fast as possible" instruction over the "as wide as possible" instruction. With this, smaller amplitudes in finger tapping recordings were common, making the separation of a ‘tap’ from a hesitation or anomaly less distinct, complicating segmentation using an algorithm. This phenomenon might explain the subtle but consistent hierarchy observed in tasks, whereby pronation-supination and hand open-close are more exaggerated movements that are less open to interpretation, demonstrated superior discriminative performance.

4.10.2 Comparison with Expert Performance

The study by Williams et al. [197] provides valuable insights into expert diagnostic performance based solely on finger-tapping recordings. In this study, 21 clinicians evaluated 133 videos, 73 from 39 individuals with idiopathic PD and 60 from 30 healthy controls. Importantly, the clinicians had no access to additional contextual information, minimising bias and establishing a benchmark for Machine Learning (ML) models trained exclusively on hand-kinematic data. The clinicians correctly identified the PD/control status in 70% of cases. Notably, all three ML models outperformed the human experts, with the MultiRocket model achieving the highest average accuracy of 81%.

These findings suggest that ML models can surpass expert performance, given these constraints. However, it is important to note that the comparison is not direct, as the datasets and assessment conditions differ between this work and the study conducted by Williams et al. Future research could explore whether this performance advantage holds when ML models and human experts are evaluated on the same dataset. Such studies could also investigate scenarios that progressively limit the information available to clinicians, such as using digital reconstructions of hand movements or focusing solely on fingertip and thumb positions, mirroring the input data used by the ML models.

The Williams et al. study [197] also highlights potential biases in MDS-UPDRS severity scoring. In which bias could stem from prior interactions with patients, potentially affecting clinical studies where control subjects are often companions of PD patients. They found that 53% of control participant videos were given an MDS-UPDRS finger tapping score greater than 0, and 25% of control participant hand videos were identified as having bradykinesia following the Modified Bradykinesia Rating Scale (MBRS) specification. The authors infer from this that finger-tapping bradykinesia may be a non-specific sign, potentially overlapping with movement changes associated with normal aging, particularly when mild. In contrast, the current study’s dataset showed a maximum of 2% of control recordings scored higher than normal. However, the misclassification rates of UPDRS scores for control recordings were 26% for InceptionTime, 19% for MiniRocket, and 18% for MultiRocket, which interestingly aligns more closely with the findings of Williams et al.

The moderate disagreement among experts viewing the same recording (ICC 0.53) reported by Williams et al. [197] underscores the challenges in achieving consistent human assessments. This variability, coupled with the common acceptance of ±1 as an 'acceptable' accuracy in automated scoring, highlights an inherent limitation in the granularity of assessments, whether performed by humans or ML models. These observations suggest that efforts to improve the reliability and consistency of PD assessments should focus on reducing disagreement among human assessors before further refinement of ML models. One potential approach could involve incorporating digital measurements and video recording into standard clinical procedure. This would allow clinicians to review more objective, quantitative data alongside their visual assessments, potentially improving consistency and accuracy.

The challenges in achieving fine-grained, consistent assessments significantly influenced the decision in the present study to focus on binary classification for severity scoring (normal vs. abnormal). This particular target of (0) vs (1,2,3,4) represents a more challenging classification task compared to the commonly investigated (0,1) vs (2,3,4) or multi-class classifications with acceptable accuracy. As such, this specific binary classification is not well reported in other studies, making direct comparisons challenging. The rationale behind this choice and its implications are further explored in the next section in context of the work by Morinan et al. [114].

4.10.3 Comparison with Feature-Based Methods

Morinan et al. employed a feature-based approach utilising a Random Forest model for multi-class and binary classification of MDS-UPDRS severity. The multi-class classification is representative of many methods in the field. Their methodology involves approximating movements with a one dimensional time series signal, from which clinically relevant feature pertaining to bradykinesia are extracted. A random forest model is then utilised to objectively quantify severity across various tasks (finger tapping, hand open-close, pronation-supination, toe tapping, leg agility). The primary motivation behind their study was to leverage the pose-estimation capabilities of advanced computer vision approaches, specifically DeepLabCut [109]. While this method offers advantages in terms of accessibility and ease of implementation, it fundamentally measures hand kinematics, making it comparable to the approach used in the current study.

A notable strength of Morinan et al.'s study is its large sample size, which addresses a common limitation in the literature. For instance Khan et al. [86] reported 94.5% accuracy for PD/control classification, but their study only included 16 PD patients and 6 controls, which gives uncertainty about the models ability to generalise. In contrast, Morinan et al.'s study included 628 PD patients across five independent sites, with left and right sides for each task, resulting in a comprehensive dataset of 10,823 ratings. This extensive dataset significantly enhances the confidence in the generalisability and reliability of their findings.

Their multi-class models, where each MDS-UPDRS score is treated as a separate class, achieved an 'acceptable accuracy' of approximately 0.85 for each of the three upper limb tasks. For their binary classification task, which distinguished between mild (0,1) and more severe (2,3,4) cases, they reported AUC-ROC scores of 0.79, 0.81, and 0.75 for finger tapping, hand open-close, and pronation-supination tasks, respectively.

While these AUC-ROC scores are comparable to or exceeded by the models in this work, it is important to note that the classification targets differ. The present study focused on distinguishing between normal (0) and any level of impairment (1,2,3,4), a more challenging task than separating mild from more severe cases.

To facilitate a more direct comparison, performance metrics for the 0 vs. rest classification were inferred from Morinan et al.'s multi-class confusion matrices:

1. Finger tapping: 15.22% of recordings were severity 0, with an F1-score of 0.83 and MCC of 0.22

2. Hand open-close: 21.75% of recordings were severity 0, with an F1-score of 0.789 and MCC of 0.21

3. Pronation-supination: 25.35% of recordings were severity 0, with an F1-score of 0.75 and MCC of 0.22

While the F1-scores are comparable to those achieved in the present study, the MCC scores are notably lower than those of the best-performing models in this work (finger tapping MCC: 0.43, hand opening and closing MCC: 0.61, pronation-supination MCC: 0.65). This discrepancy can be attributed to the imbalanced nature of Morinan et al.'s dataset, of which is accounted for by MCC metric.

4.10.4 Comparison to End-to-End Approaches

The study by Gao et al. [56] represents the current reported state-of-the-art in differentiating PD patients from healthy controls HC using the finger-tapping task. Their approach achieved remarkable AUC-ROC scores of 0.959 and 0.976 for the left and right hands respectively. These results notably surpass the performance of the best model for finger-tapping in this work, MultiRocket, which achieved an average AUC-ROC of 0.87 for the same classification task.

Gao et al. employed Cartesian Genetic Programming, an evolutionary algorithm that generates a symbolic model applied to a 30-second 1-D acceleration time-series capturing finger-thumb separation. The extended recording duration may better capture fatigue effects, contributing to the improved performance compared to the 5-second recordings used in this work. Moreover, the model was trained on a UK dataset and evaluated on a separate Chinese cohort, providing empirical evidence of its generalisability—a key strength, while this thesis relies on cross-validation for performance estimates.

Several methodological differences, however, may help explain the observed performance gap. Firstly, the study utilises the larger of two scores from repeated tasks for each hand, which may give an advantage in capturing more pronounced symptoms. Secondly, with many PD subjects in early stages of the disease, where symptoms present unilaterally, and given that all participants were right-handed, the right-hand performance is expected to be higher. This may partly explain the stronger results observed for the right hand. Finally, the study excluded 10 PD patients with zero bradykinesia scores to focus on differentiating varying levels of bradykinesia severity. While this approach aligns with their objective, it contrasts with this work's inclusion of all cases, including those with zero severity scores, which are more challenging to classify.

4.10.5 Limitations and Future Work

4.10.5.1 Comparison to Traditional ML Methods

This work would have benefited from a comparison to feature-based methods on this dataset. Nevertheless, significant challenges arose in developing an approach to achieving this. The segmentation of kinematic signals, such as identifying individual taps in a finger tapping time-series, is a crucial step in the pre-processing pipeline for feature-based machine learning studies in this field. Despite its importance, many published works that implement segmentation as part of their methodology provide limited details on their specific implementation of peak finding algorithms. This lack of detailed reporting creates challenges for reproducibility and standardisation in the field.

Notably, studies such as [94, 19, 77, 21, 118, 117, 119, 164, 181] mention the use of peak detection algorithms in their methods but either do not provide comprehensive information on algorithms used or do not justify parameters used. This gap underscores the necessity for a more transparent and objective approach to signal segmentation in the context of MDS-UPDRS motor assessments.

The subjective nature of the MDS-UPDRS motor assessments themselves may contribute to this variability in implementation and reporting. For instance, in the finger tapping task, subjects are instructed to "tap the index finger on the thumb 10 times as quickly AND as big as possible." Examiners are expected to look for interruptions and decrements in amplitude, but these criteria are not precisely defined, leaving room for interpretation in both clinical and automated assessments.

The application presented in Appendix B aims to tackle these challenges by developing a dataset to drive a data-driven method for optimizing the parameters of the find_peaks signal. This approach seeks to enhance the objectivity and reproducibility of kinematic signal analysis in motor impairment assessments, addressing the variability in implementation and reporting observed in previous studies.

4.10.5.2 Exploration of an Anti-Assessment

A recurring concern in applying machine learning models in clinical settings is the inherent lack of transparency—often referred to as the "black box" problem. This issue is particularly pressing in healthcare, where decisions informed by models must be trusted by clinicians and patients alike [138]. Researchers must also exercise caution when interpreting published results, especially when the mechanisms behind model decision-making remain opaque [104].

One approach to enhancing trust and model robustness involves introducing a form of anti-assessment. The objective of this anti-assessment would be to design tasks or conditions where, theoretically, the model learns nothing of discriminatory value. Achieving this is especially challenging in the context of PD, given the disease's hallmark motor impairments. However, the remarkable phenomenon where PD patients can cycle effortlessly despite severe gait disturbances, including freezing of gait [187], suggests that under certain highly constrained conditions, motor tasks can be performed with minimal manifestation of typical impairments.

An upper-limb equivalent might involve developing a goal-oriented, highly structured task with maximum sensory feedback, minimising observable motor differences between individuals with PD and healthy controls. One potential task could involve subjects rhythmically moving their arm like a metronome along a flat surface, pivoting on the elbow and oscillating between two fixed points. By integrating rich sensory feedback—both visual and auditory—the task could ”normalise” movement to the extent that no meaningful discriminatory features emerge.

The aim would be to assess the model's performance under these conditions to determine if it indeed learns nothing of value. If successful, this approach could enhance trust in models by demonstrating their ability to distinguish between meaningful and non-meaningful tasks. This could be especially relevant for tasks that have not yet been formally investigated, such as the Luria test—a sequence of hand gestures used for identifying cognitive impairment [194]—where, to the best of the author's knowledge, no digitised approaches have been published.

4.11 Conclusion

The results of this study demonstrate that end-to-end time series classification models, particularly InceptionTime, MiniRocket, and MultiRocket, are effective in analysing upper limb movement data for PD diagnosis and detection of slight bradykinesia. These models achieved performance comparable to, and in some cases surpassing, traditional feature-based approaches and human expert assessment, with the added advantage of not requiring extensive feature engineering or domain expertise.

A key strength of these models lies in their ability to learn directly from raw positional data, offering flexibility in adapting to various data sources and potentially identifying complex patterns that might be overlooked by summary measures. This adaptability is particularly valuable as wearable sensor technology advances and becomes more prevalent in healthcare applications.

Future work should focus on further improving model interpretability to enhance clinical applicability and potentially uncover new insights into PD manifestation and progression. Additionally, investigating the models' performance on larger, more diverse datasets and exploring their applicability to other movement disorders could further validate and extend the utility of this approach in clinical settings.

Chapter 5

Semantic Segmentation of Neuropsychological Figures

Digitisation of traditional neuropsychological pen-and-paper assessments provides a means to amplify their diagnostic utility. Features derived from the precise recording of temporal pen dynamics (position, pose and pressure) have demonstrated significant promise as biomarkers in PD diagnosis [203, 185, 201, 31, 52].

However, deriving features that characterise executive function, specifically related to organisation and planning, requires the semantic labelling of strokes. Existing tools for the semantic segmentation of the ROCF, a widely used tool for such analyses, remain suboptimal for routine clinical use. This is primarily due to the requirement for time-intensive manual labelling. A similar problem has been addressed in the semantic segmentation of digital images, where deep learning models perform the initial labelling, which is subsequently refined manually [162].

In this chapter, a modified U-Net architecture [156] is evaluated for the automatic semantic segmentation of a simplified ROCF figure. Central to this approach is the novel exploration of feature embeddings, where pen dynamics such as pressure, time and tilt, are represented via pixel intensities while retaining their spatial position. This multidimensional representation of pen dynamics aims to further improve segmentation of digital drawings.

5.1 Background

The Rey-Osterrieth Complex Figure [151, 131], composed of hierarchically structured abstract geometric shapes, serves as a prominent non-verbal tool in neuropsychology [171]. Reproduction of the figure from example and recall, provides a cognitive snapshot of visuospatial constructional ability, visuospatial memory, and processing speed [111]. Since it's introduction in the 1940's the notion and utility of complex figures has been expanded upon. With alternatives such as the Taylor figure that mitigates practise effects [183], and simpler versions such as the Benson Figure or Geriatric Figure for cognitively impaired and elderly populations [143, 142].

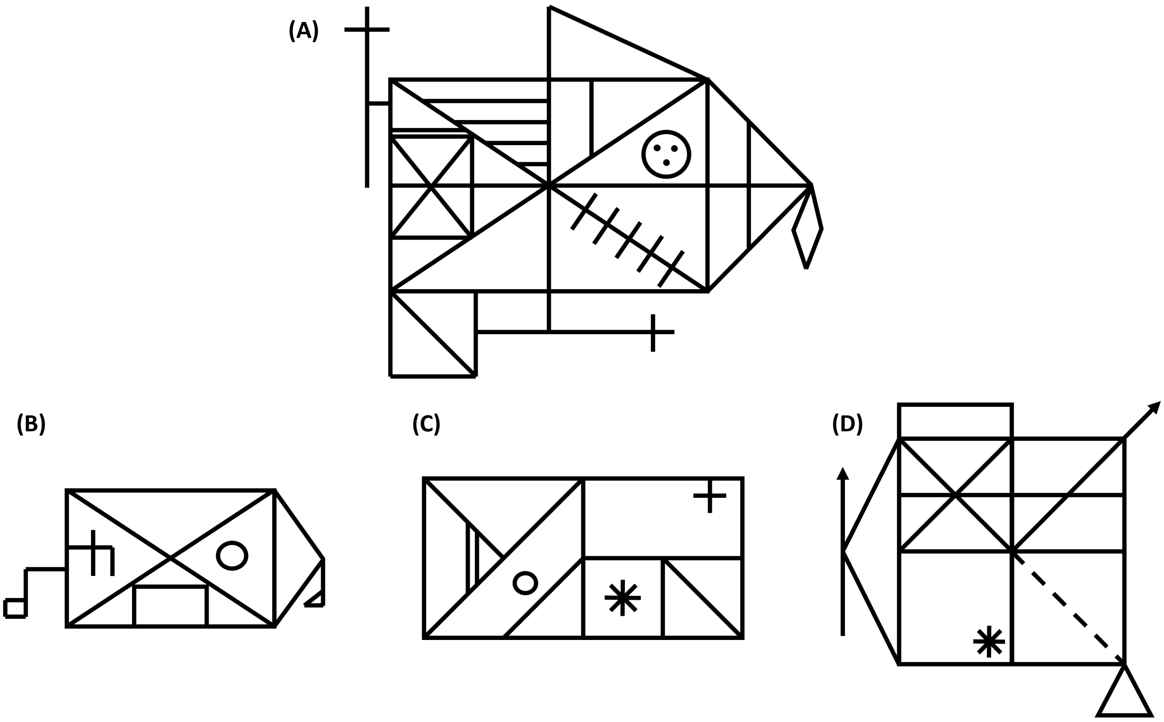


Figure 5.1: Rey-Osterietth Complex Figure [131] (A) and simplified versions, the Benson Figure [143] (B), the OCS-plus figure [43] (C), and the simplified Taylor-Figure [42] (D).

The conventional method for scoring the ROCF is based on the Osterietth system [131]. In this, the figure is divided into into 18 components, which are individually assessed based on three criteria: accuracy, completeness and placement. This method is straightforward and allows for post-hoc evaluation, but does not provide information regarding the organisational strategy employed during the drawing process.

To address the limitations of the Osterietth method, the Boston Qualitative Scoring System (BQSS) [180] was developed. This system introduces five scores related to executive functioning (planning, fragmentation, neatness, preservation, and organisation), derived from hierarchical groupings of the figure (configurable, clusters and details). A study by Scarpina et al. [166] utilised the BQSS to differentiate between PD patients and healthy controls. The results indicated that lower planning and neatness scores associated with PD patients were associated with executive function deficits, notably in planning and impulsivity, rather than impaired visuospatial constructional ability. However, despite the utility of the BQSS, it is constrained by administration challenges and trade-offs.

Two strategies are employed for the recording of pen strokes, the flowchart method and pen-switching method [176]. In the flowchart method, the examiner annotates a reference figure, with arrows to indicate the direction of each stroke, and numbers to show the order. In comparison, the pen-switching method, suggested by Rey [152], denotes the ordering of strokes via different coloured pens. In this, pens are switched at predefined intervals, usually when a subject has completed a section of the drawing. Somerville found that although the administration time for both methods was the same, the flow-chart method took longer to score [176]. Overall, a trade-off must be made between the granularity of detail captured and labour required from the examiner.

The recent study by Petilli et al. [139] signifies a notable advancement in this domain. Their Tablet-based Rey Complex Figure copy task (T-RCF) offers a comprehensive digital solution, that fulfils the objectives previously targeted by the pen switching and flowchart method. By extracting a wide range of indices, including spatial, procedural and kinematic aspects, the digital methodology affords a more nuanced and complete evaluation of drawing abilities. However, this is contingent upon the manual segmentation of the image, a limitation highlighted by the authors, and one which this work aims to address.

Previous automated segmentation approaches of complex figures have focused on the task of automated scoring. Early work by Canham et al. [24], used algorithms to search for the identification of basic shapes (triangles, rectangles, prisms and simple lines), representing an offline drawing as an attributed relational graph (this facilitated the representation of collinear lines in the geometric shapes). While the method could identify only 6 basic patterns, their method achieved 99.3% accuracy, in locating these sections. More recently Webb et al. [193], have also found success with similar heuristics for the OCS Figure Copy Task, although noted that generalisability to other figure tasks is not guaranteed without further algorithmic development.

In the context of sketch segmentation literature (as well as the broader domain of computer vision), deep learning approaches have emerged as state of the art [100, 198, 148, 147], surpassing conventional heuristic methods. Notably the approach by Li et al. [100], is through the rasterisation of digital drawings followed by the application of a slightly modified U-Net architecture.

U-Net is a convolutional neural network architecture tailored for semantic segmentation tasks [156]. The efficacy of U-Net is attributed its encoder-decoder structure, with skip connections to preserve information that may otherwise be lost of during the encoding process. In this methodology a digital image serves as the input, and the network outputs a mask, whereby each pixel is classified according to its semantic content. This rasterisation and U-Net based approach has been applied to other neuropsychological drawing tasks, such as the clock drawing test [134], and the interlocking pentagon task [133]. In these applications, the generated mask facilitates automated scoring.

Similar to the post-hoc scoring methods used for the ROCF, the rasterisation of the drawing used in previous approaches fails to capture pen dynamics. Allisa et al. [5] addressed this gap by encoding pressure information via pixel intensities in greyscale digitisations of interlocking pentagons and necker cubes. Their approach improved the classification performance in discriminating between PD and healthy controls. Specifically, they extended the conventional binary (black and white) representation to a greyscale image. The grey information was generated by scaling the pressure values of $[0, 1]$ to $[0 - 254]$. In addition in air trajectories were considered, in this the grey value was set to 0, 255 was used to indicate the absence of pen data. They found that the additional encoding improved the accuracy and stability of the classifiers training.

The objectives of this present work are twofold. The first is to evaluate the U-Net architecture for the semantic segmentation of a simplified complex figure task, thereby laying the groundwork for future application to the ROCF and other complex figures, without the requirement for handcrafted heuristics. The second objective is accompanied with a specific hypothesis, that the encoding of pen dynamics, as done by Allisa et al. [5], extended to three dimensions to also include tilt and temporal information, will be more effective than a binary representation for semantic segmentation tasks. This is based on the presumption that providing additional information about pen dynamics will enhance the models ability to discriminate between overlapping strokes, a task that becomes particularly challenging in a binary images where, no information is retained about occluded sections, and in lower resolutions where the boundaries between lines becomes increasingly difficult to discern. Figure 5.2 is provided to illustrate this encoding, and give context towards the structure of the methodology.

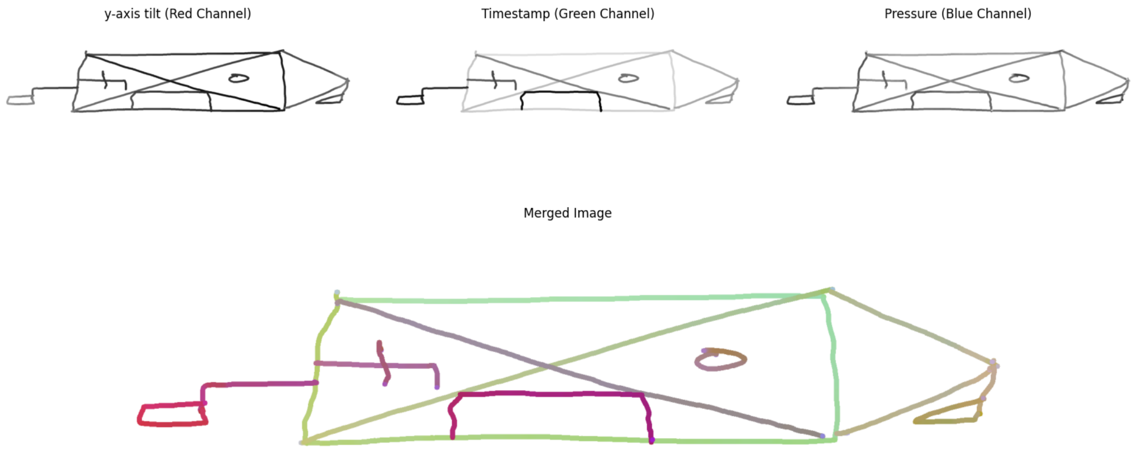


Figure 5.2: Multi-channel representation of a pen-dynamics from a digital drawing, illustrating the individual contributions of y-axis tilt (Red channel), time (Green channel) and pressure (Blue channel), alongside the resulting image obtained from concatenating these channels. Each subplot is displayed in greyscale, and the merged image

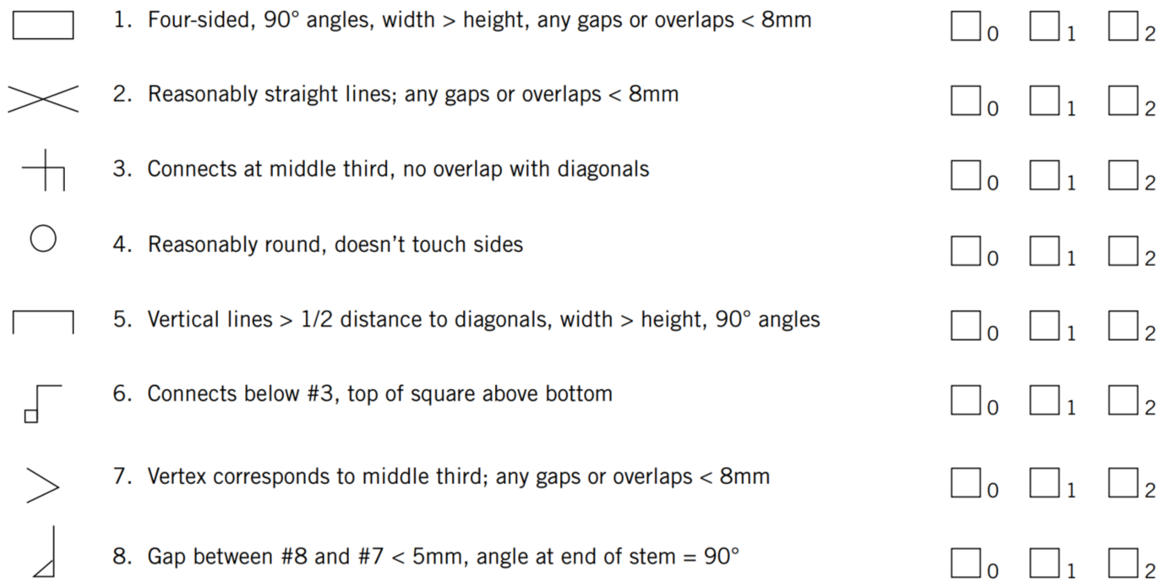
5.2 Methodology

The semantic segmentation of the Benson Complex Figure, a simplified version of the ROCF developed by Frank Benson M.D., is investigated. Possin et al. [143] found that the copy task was adequate to elicit differences between subjects with Alzheimer's disease and the behavioural variant of Frontotemporal dementia. Similiar to Webb [193], a simplified figure serves as an exploration into automated segmentation techniques that can then be applied to the ROCF, and thus the Benson Figure is an appropriate starting point. The goal is to classify each element of the figure, described by the National Alzheimer's Coordinating Center - Uniform Data Set (NACC-UDS) [116]. The NACC-UDS provides standardised assessment tools for Alzheimer's disease research.

Figure 5.3 presents the elements and scoring criteria for the Benson Complex Figure, taken from the NACC scoring criteria. The eight elements of the figure correspond to their respective number found in the scoring criteria.

5.2.1 Dataset Acquisition

The digital drawings used in this work, are sourced from the study "A novel diagnostic devices for the objective diagnosis of Parkinson's disease with and without dementia" [39]. The dataset comprises information acquired from 58 patients diagnosed with PD, as per the Queen Square Brain Bank Criteria, by specialist consultants in neurology clinics. Additionally, the dataset includes 29 age-matched healthy controls, who were either spouses or friends of the



1. Four-sided, 90° angles, width > height, any gaps or overlaps < 8mm □0 □1 □2

2. Reasonably straight lines; any gaps or overlaps < 8mm □0 □1 □2

3. Connects at middle third, no overlap with diagonals □0 □1 □2

4. Reasonably round, doesn't touch sides □0 □1 □2

5. Vertical lines > 1/2 distance to diagonals, width > height, 90° angles □0 □1 □2

6. Connects below #3, top of square above bottom □0 □1 □2

7. Vertex corresponds to middle third; any gaps or overlaps < 8mm □0 □1 □2

8. Gap between #8 and #7 < 5mm, angle at end of stem = 90° □0 □1 □2

Figure 5.3: Elements and scoring criteria for the Benson Complex Figure, taken from the NACC, scoring criteria.

PD patients. All data were collected by clinicians at Leeds Teaching Hospitals NHS Trust (LTHT). The study received National Regional Ethics Service approval (10/H1308/5), and local and research and development approval from LTHT (UI10/9232).

Future work will investigate how non-semantic cognitive features compare with features that correspond to organisation. Cognitive features from [145] include number of strokes, sketching time, stroke distance, dureation, average pressure, average velocity, velocity variation, number of pauses, average pause duration, the ratio between sketching duration and pausing duration, and average lift duration. Organisational features from [139] include:

1. The level of priority given to the most relevant unit of the figure (i.e. the base rectangle).

2. The level of priority given to the less relevant unit of the figure (i.e., the inner details, with the first feature they can give a global vs local measure)

3. The number of time the reproduction of a component of the figure was interrupted to reproduce elements belonging to other units.

These features will later be applied to discriminating between levels of cognitive impairment. Together with this segmentation approach, they will facilitate a comprehensive solution for the quick extraction of clinically relevant features.

5.2.1.1 Protocol

Subjects were instructed to copy the Benson Complex Figure based on a reference image, without any time constraint. Subsequent scoring was conducted by trained assessors post-hoc, evaluating both the accuracy and placement of each element. The maximum attainable score was 17 points. [39].

For data capture, the Wacom Intuos5 Touch L PTH850 graphics tablet was selected due to its non intrusive nature in recording graphmotor data. When overlaid with a sheet of paper, the inking stylus provides an experience equivalent to a traditional pen. Figure 5.4 illustrates the experimental setup.

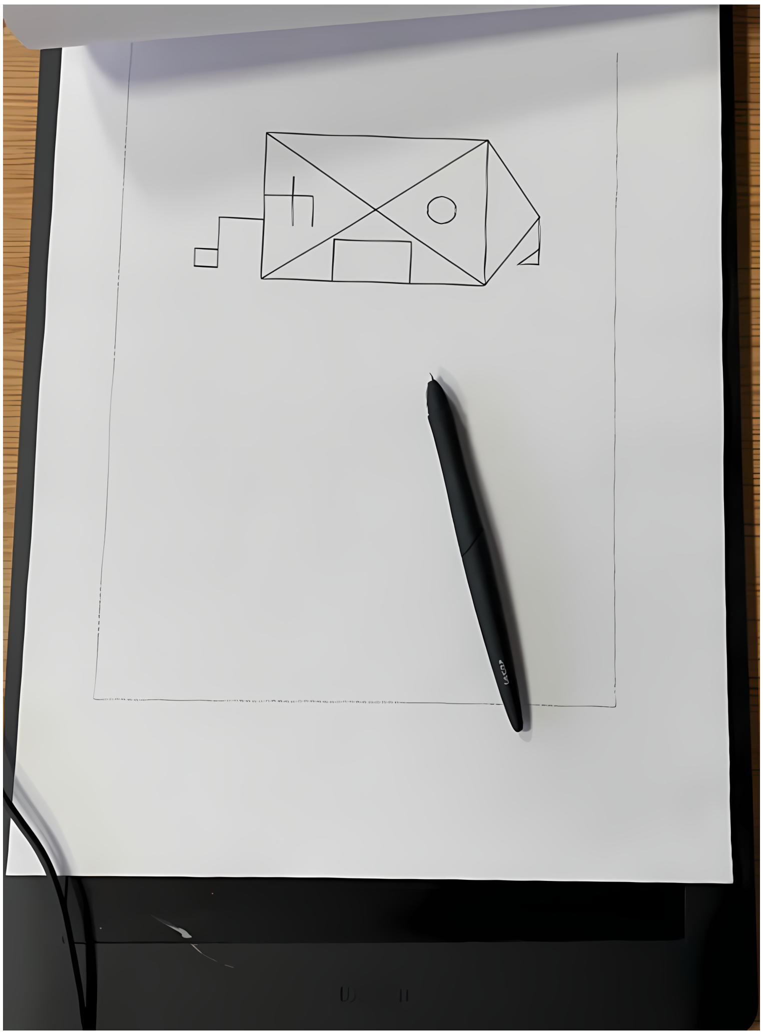


Figure 5.4: The wacom graphics tablet, with overlaid assessment sheet

Wacom graphics tablets operate using the principle of Electromagnetic Resonance (EMR) [192], whcih allows for precise stylus tracking without the need for batteries or wires in the stylus. At a sampling rate of 200Hz, the tablet is able to locate the position of the stylus to an accuracy of ±0.25 mm and the a tilt measurement in ±60°, on the x and y axes. Moreover, the stylus is able to report 2048 discrete pressure values and button presses. The active area of the tablet is 32.5 x 20.3 cm, comparable to the size of a A4 sheet of paper. One notable feature of EMR technology is the ability to measure the position of the stylus while hovering above the surface of the tablet. This also permits the integration of an LCD screen.

The JTablet 2.0 API was used in the data acquisition software. For each sample, normalised x and y positions, x and y tilt and pressure values were recorded. Timestamps were appended upon receiving each sample.

5.3 Data Preprocessing

5.3.1 Raw Sketch Data

The following notation will be used for the pen data:

$$px, py : \text{coordinates} \tag{5.1}$$

$$tx, ty : \text{tilt angles} \tag{5.2}$$

$$pr : \text{pressure} \tag{5.3}$$

$$ts : \text{timestamp} \tag{5.4}$$

The raw drawing data D, comes in the form of multivariate time series data.

$$D = \begin{pmatrix} px_0 & py_0 & tx_0 & ty_0 & pr_0 & ts_0 \\ px_1 & py_1 & tx_1 & ty_1 & pr_1 & ts_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ px_{n-1} & py_{n-1} & tx_{n-1} & ty_{n-1} & pr_{n-1} & ts_{n-1} \end{pmatrix} \tag{5.5}$$

Where x_i represents the x coordinate of the ith sample point within the drawing, with $0 \leq i \leq n$. The ith sample can be referenced by s_i, which corresponds to $(px_i, py_i, tx_i, ty_i, pr_i, ts_i)$.

5.3.2 Sample Segmentation

The graphics tablet is event driven, as such it cannot be assumed that samples are received are at regular intervals. This irregularity is illustrated in Fig 5.5, where linear interpolation between non-continuous samples results in sharp, erroneous trajectories.

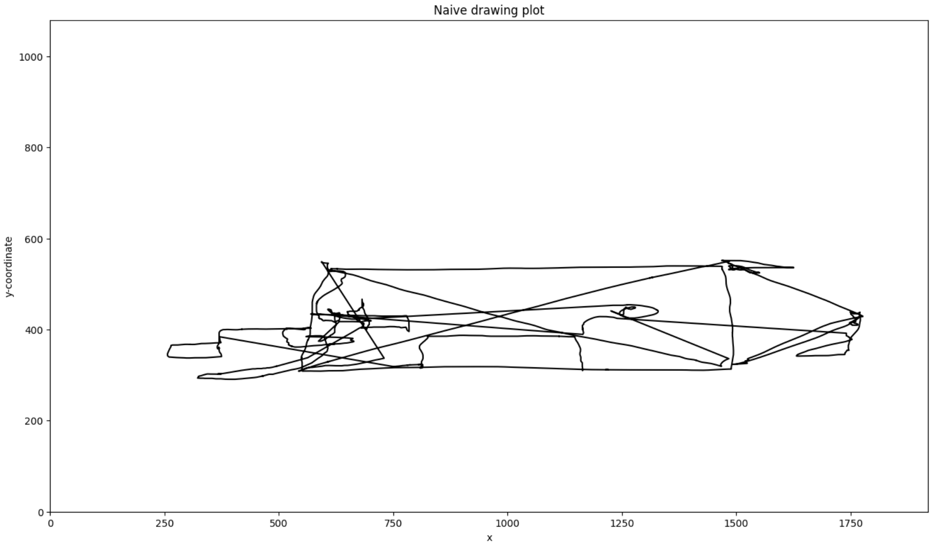


Figure 5.5: A naive recreation of the drawing, plotting the coordinates of each sample and linearly interpolating between neighbouring samples.

To achieve a more accurate representation, samples are grouped into down strokes (periods where the pen is in contact with the surface) and air strokes (periods where the pen is not in contact). This grouping necessitates the inference of two types of events from the data:

1. **Contact Events**: Characterised by transitions between stylus contact and non-contact states on the tablet surface.
   - **Pen Up Event:** $pr_i = 0$ and $pr_{i-1} > 0$
   - **Pen Down Event:** $pr_i > 0$ and $pr_{i-1} = 0$
2. **Proximity Events**: Governed by stylus entry or exit of the tablet's active area or hover range. This event is inferred when the time elapsed between neighbouring samples exceeds a threshold.
   - **Exiting Event:** $ts_{i+1} - ts_i \geq threshold$
   - **Entering Event:** $ts_i - ts_{i-1} \geq threshold$

For this work the threshold is set to $30ms$ (experimentally chosen), as discussed in Section 5.3.2.1 the precision to which missed samples can be inferred is hindered by the recording software. It should be noted that Wacom tablets do emit proximity events, and it is recommended that future work utilises this event instead.

A **stroke** is defined as a temporally contiguous sequence of samples where the pen is either continuously in contact with the surface of the tablet (*down stroke*) or hovering above it (*air stroke*). Specifically:

- **Down stroke**: Begins with either the first sample of the drawing, a pen down event, or an entering event. It ends with either the last sample of a drawing, a pen up event or an exiting event.
- **Air stroke**: Begins with either a pen-up event or an entering event. It ends with a pen down event or exiting event.

In this study, strokes with a length less than 0.1 cm were deemed unlikely to represent relevant movements in the reproduction of the Benson Figure. Consequently, these strokes were classified as noise and excluded from further analysis. Given these grouping and the removal of noise, an accurate representation of the data is presented in Figure 5.6.

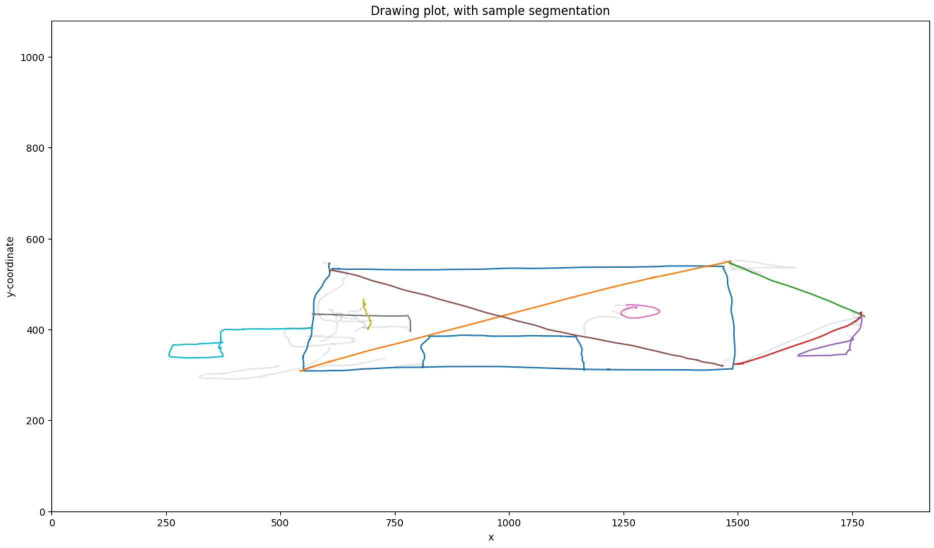


Figure 5.6: Drawing with samples segmented into strokes. Down strokes are are assigned a unique colour. Air strokes, are represented by transparent black lines.

The limited height range of the tablet results in frequent entering and exiting events when the stylus is hovering. Notably, previous studies utilising air strokes in their analyses have not addressed the role of proximity events in their methodologies [201, 158, 5]. While this omission may have less impact on analyses of handwriting or simpler figures, where the stylus is expected to be in close proximity to the surface of the tablet; for the current dataset, the detection of these proximity events has been crucial for correcting an error in the timestamps.

5.3.2.1 Timestamps

A noticeable discrepancy in the calculated sampling rate of the recorded timestamps was observed, as shown in Figure 5.7. The sampling rate of down strokes was much lower than the expected 200 Hz and varied considerably across strokes. This variation was not as prominent in air strokes. Analysis of the acquisition software revealed that for each sample received, the software added a timestamp based on the current system clock, and for samples received during a down stroke, a new frame was rendered, introducing a delay which compounds over the length of the stroke. In contrast, air strokes did not encounter this issue, as they were not rendered.

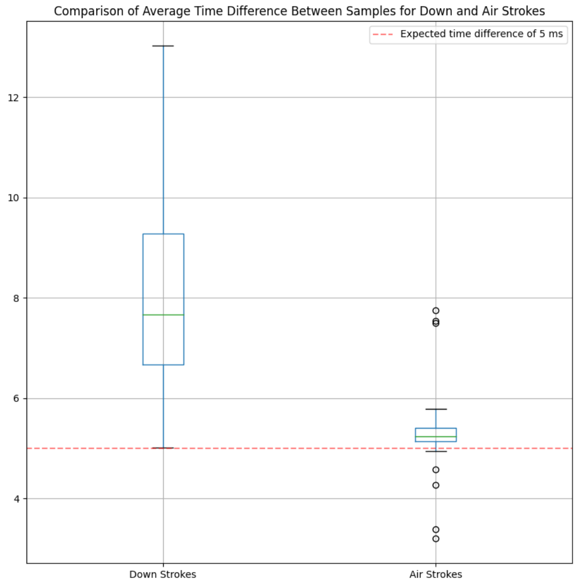


Figure 5.7: Boxplot illustrating the average time differences between samples for both down strokes (pen in contact with the surface) and air strokes (pen off the surface, but still in range) for a recording. The red dashed line indicates the expected average time difference of 5 ms at a sampling rate of 200 Hz. This visualisation aims to exhibit how rendering affects the latency in capturing timestamps.

In an attempt to correct this inconsistency, it was first assumed that the buffer was sufficient to prevent data loss, thus with each stroke constantly reporting events a constant sampling rate of 200 Hz was also assumed for each stroke. For each lift event, if an exiting event was detected, the time gap until the next entering event was calculated and then added to all subsequent timestamps, including the entering event itself. This adjustment ensures that the timing aligns with the expected sampling rate despite delays introduced during down strokes. This correction was required as the timestamps of the samples is encoded in the rasterised image. Accurate timing is also crucial for the analysis of drawing speed, hesitations, and other temporal features that may be indicative of cognitive function in neuropsychological assessments.

5.3.3 Stroke segments

Any single stroke can be arbitrarily complex, and may correspond to multiple elements of the Benson figure. For instance, it could be possible that the entire figure is drawn with a single stroke. This presents an annotation challenge, as any stroke may have to be split into stroke segments, in order to assign correct labels. For the current dataset, approximately 30% of the drawings contained strokes that met this criterion. Petelli et al. addresses this in their T-RCF software through the use of manually added break points to a stroke [139].

The current work explores an automated solution to this challenge. The Ramer-Douglas-Peucker (RDP) algorithm, a polyline simplification technique, has been utilised to achieve this goal [1]. The RDP algorithm recursively reduces the number of samples in a stroke until a specified threshold, ϵ is met. The pseudocode for this algorithm is presented in Appendix A. Points within a stroke are grouped between these remaining points. Consequently, all points within a stroke segment belongs to the same class.

A primary limitation of this method is the necessity for empirical determination of the epsilon value. The optimal value is dependent on the relative size and level of detail in the drawing, which may introduce variability in the segmentation process. An epsilon value of 10 was deemed appropriate. This value was applied after scaling the drawings: the x-axis was scaled to meet the width of a 1:1 aspect ratio square, denoted as W, and the y-axis was multiplied by the same scalar ($\frac{W}{\text{original width}}$) to preserve the original drawing's aspect ratio. This scaling and segmentation process is illustrated in Figure 5.8.

5.3.3.1 Sketch Annotation

Due to the absence of suitable existing tools for sketch annotation meeting the specifications required for this research, a bespoke sketch annotation interface was developed. This custom Graphical User Interface (GUI) was developed specifically for this purpose, as illustrated in Figure 5.9, not only serves the immediate requirements of this thesis but also contributes a novel utility to the broader academic field.

The GUI allows for efficient and precise data annotation by rendering sketch data on an

[1] This approach is used in [193], albeit not for the purpose of specifically labelling segments

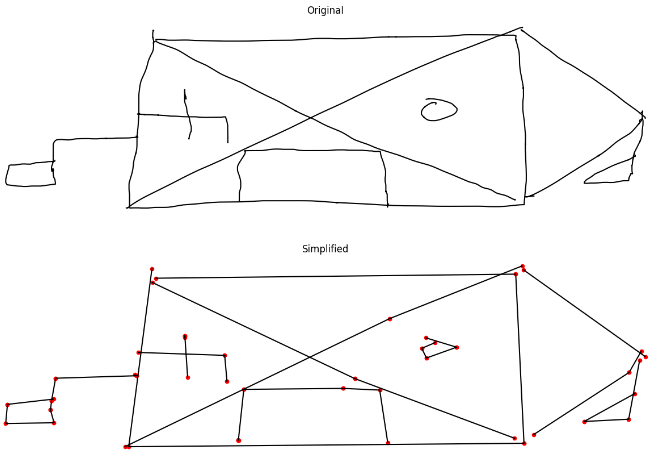


Figure 5.8: Before and after polyline simplification using the Ramer-Douglas-Puecker algorithm. The top panels displays the original down strokes with all of the recorded samples. The bottom panel shows the line segments constructed from the points selected, (just 1% of the original points).

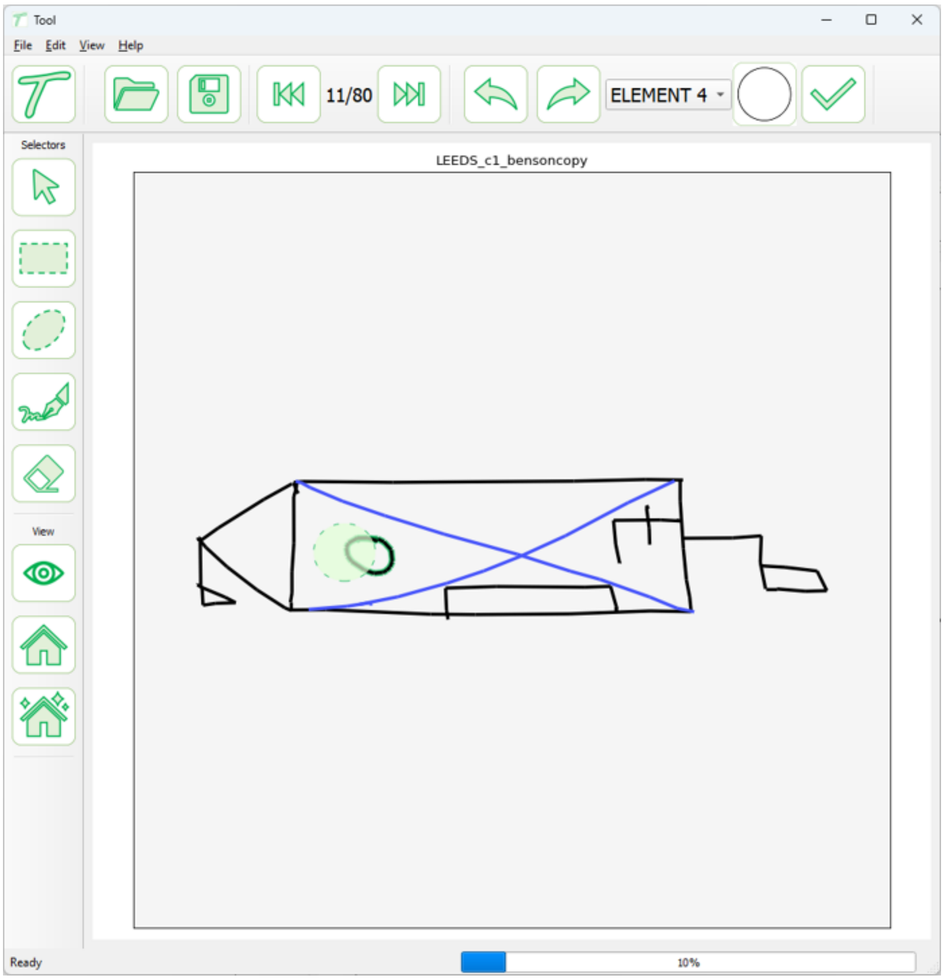


Figure 5.9: Custom-developed sketch labelling software employed for stroke segment classification.

interactive canvas. The interface facilitates the selection and categorical labeling of individual stroke segments. The labels can be modified by the user, but for this work are derived from the eight elements within the NACC-UDS Benson Figure scoring criteria 5.3.

Within this interface, each stroke segment can be selected and categorically labelled. However, not all stroke segments align neatly with these criteria; segments that defy classification are accordingly labelled as "UNKNOWN", effectively expanding the label set to nine distinct categories.

To facilitate the annotation process, two selection modalities have been integrated into the GUI:

1. **Picker Tool**: Allows for individual selection of stroke segments for manual classification.
2. **Shape Tool**: Enables bulk selection of stroke segments within a defined geometric boundary, either rectangular or elliptical in shape.

The software was implemented in Python [189], leveraging the PyQT GUI framework [146] for GUI implementation and Matplotlib [76] for rendering the sketch data on the canvas.

5.4 Dataset Generation

5.4.1 Rasterisation Process and Segmentation Map Generation

The rasterisation process converts the timeseries stroke data into a 3D RGB image format. This format was chosen for its compatibility with pre-trained models and its ability to encode multiple features simultaneously, making it convenient to work with the chosen machine learning library fastai [73]. The procedure encompasses the following steps:

5.4.1.1 Input Image Rasterisation

The rasterisation process converts the timeseries stroke data into a image format for the neural network. This was chosen as a 3D RGB image, as it is convenient to work with the chosen machine learning library fastai [73], which has pre-trained models for this image format. The procedure encompasses the following steps:

1. **Initialisation of Array**: An square array is initialised with zeros, with dimensions corresponding to the desired resolution (e.g. 512x512 or 128x128). Pixel values of zero are reserved for the background class.

2. **Coordinate Scaling**: The coordinates of the drawing are normalised such that the largest axis (uniformly the x-axis in this study) spans a range [0 to 1]. The secondary axis is subsequently translated to be centrally aligned, and scaled to the range [0, 1] such that the aspect ratio of the original drawing is preserved.

3. **Padding Addition**: A padding of 5% is applied to all sides of the image. This is implemented by first scaling to fit the image within 90% of the image dimensions and then centering the image. This ensures that there is no clipping at the edges.

4. **Feature Scaling**: Characteristics such as pressure, tilt, and timestamps are linearly scaled to fit within the integer range [1, 255].

5. **Stroke Rasterisation**: Each stroke is drawn on the image in chronological order using the following process:

   (a) Each sample point in the stroke is represented by its spatial coordinates, a thickness value, and a pixel value (corresponding to pressure, tilt, timestamp or constant). For this study, the thickness is set to 1 pixel, resulting in thin lines. Larger thickness values would create disks, resulting in thicker strokes .

   (b) Between each neighbouring two points a line is drawn. This is done through linear interpolation. Additionally, the value assigned to each pixel is also interpolated along the line. As such for dynamic features, the intensity varies along the stroke.

   (c) When strokes overlap, the most recent stroke takes precedence, mimicking the natural layering effect of drawing.

6. **Channel Assignment**: Features are assigned individually to one of the three channels of the output image based on the specific representation being generated (as per Table 5.1).

7. **Image Generation**: The three channels are combined into a single RGB image for input to the U-Net model.

This rasterisation process effectively translates the time-series stroke data into a spatial image representation. It preserves both the visual characteristics of the strokes and encodes the dynamic information of the drawing process, providing a rich input for subsequent analysis or machine learning tasks.

5.4.1.2 Segmentation Map Generation

Alongside the input image, a corresponding segmentation map is generated. This map serves as the ground truth for training the model and evaluating its performance, enabling the model to learn the correspondence between pen dynamics and figure elements. The process includes:

1. **Initialisation of Array**: A zero-filled array matching the input image dimensions is created.

2. **Stroke Segment Mapping**: Each stroke segment is assigned a unique identifier.

3. **Label**: Each stroke segment is labelled according to the element of the Benson Figure it represents. This is done via manual annotation.

4. **Rasterisation**: Similar to the input image process, stroke segments are rasterised onto the segmentation map. The same coordinate scaling and 5% padding are applied. Instead of feature values, each pixel is instead assigned the label of the corresponding stroke segment.

5. **Background and Unknown Labelling**: Pixels not covered by any stroke segment are labelled as background. Stroke segments manually labelled as ”UNKNOWN” during the annotation process retain this label in the segmentation map.

This dual process of input image rasterisation and segmentation map generation creates paired data suitable for training and evaluating the semantic segmentation model. The input images encode both spatial and dynamic information about the drawing process, while the segmentation maps provide the ground truth labels for each pixel.

Additionally, another segmentation map that corresponds to the unique segment identifier for each pixel is also generated. This follows the same process as previously described, with the exception that the segment identifier isn’t swapped out for a class label. This additional map is used after inference for recovering the predictions over the entire stroke segment, which are later used for making a single prediction for that stroke segment. This in turn recovers the original annotation made.

5.5 Evaluation of Pen Dynamics

To assess the importance of different pen dynamics in the semantic segmentation task, four distinct three-channel input image data representations were evaluated, see Table 5.1. These combinations were carefully selected to explore various aspects of the drawing process and their potential impact on segmentation performance.

Combination	Channel 1	Channel 2	Channel 3
Base Comparison	constant	constant	constant
Combination 1	x tilt	y tilt	pressure
Combination 2	x tilt	y tilt	time
Combination 3	pressure	time	constant

Table 5.1: Summary of Pen Dynamic Permutations

The selection of permutations for pen-dynamics aims to examine the relative importance of potential synergies between the pen-dynamics and the components of the Benson Figure. Each combination was designed to capture different aspects of the drawing process:

- **Base comparison**: A constant value across all three channels serves as a control, representing the spatial information of the drawing without any additional pen dynamics.

- **Combination 1**: This combination captures the three-dimensional interaction of the pen with the drawing surface. Tilt information may help discriminate between close stroke segments. For example the sides of the base rectangle will differ in orientation that the other elements. The force applied during different parts of the drawing, indicates where a stroke started and ended.

- **Combination 2**: By replacing pressure with time, this combination allows for a more granular image of how pen orientation changes throughout the drawing process.

- **Combination 3**: This combination isolates pressure and time, potentially revealing patterns in the intensity and pacing of drawing strokes. The constant value in the third channel serves as a control.

By comparing the performance of these different combinations, it can be assessed whether certain pen dynamics are more informative for the segmentation task, and also if this varies with image resolution.

5.6 Evaluation of Resolution

Notably, rasterisation is an inherently lossy procedure. While it transforms stroke data into a form suitable for computational models, this discretisation may result in a loss of fine-grained information present in the original, much higher resolution data. Consequently, a trade-off exists between memory requirements and stroke segment representation, as illustrated by Figure 5.10.

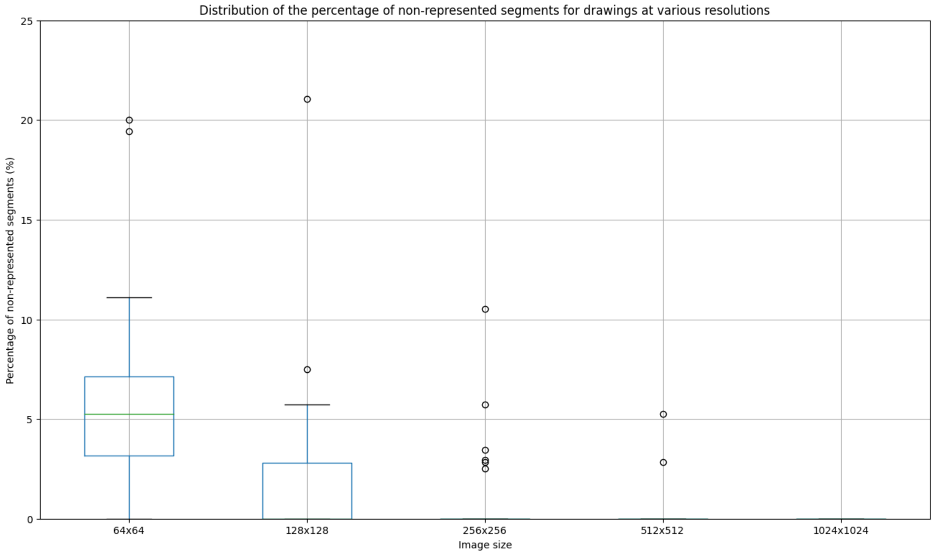


Figure 5.10: Distributions of non represented stroke segments over the data set for various resolutions

Among the 80 images in the dataset, when rendered at 512 x 512 resolution, only 2 images have stroke segments that are not represented. Conversely, images rendered at 128x128 resolution begin to lose significant stroke segment representation. To address this issue, a method must be developed to infer the class of these missing segments, ensuring a complete classification of all stroke segments regardless of resolution.

Additionally, crucial spatial details are lost at the lower 128x128 resolution. This loss of detail could have a more significant impact on more intricate figures such as the ROCF.

Figure 5.11 visually demonstrates the impact of resolution on image quality and information retention. The comparison between 128x128 and 512x512 resolutions illustrates the trade-off between computational efficiency and detail preservation, highlighting the importance of resolution selection in maintaining the integrity of the original drawing's features.

The choice between 128x128 and 512x512 resolutions for this study was made to explore this trade-off, with the higher resolution potentially preserving more information at the cost of increased computational demands.

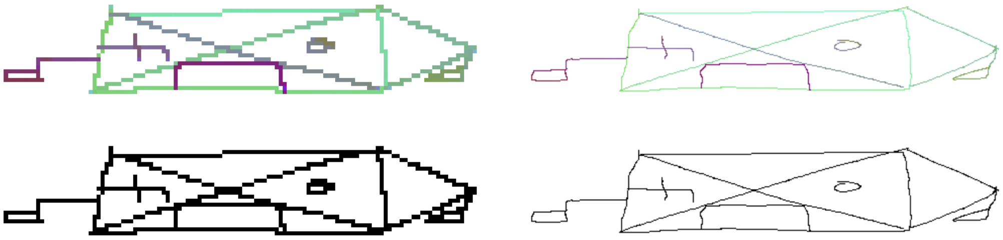


Figure 5.11: Comparison of resolution and encoding effects. Drawings in the first column are rendered at a resolution of 128x128, while those in the second are at 512x512. The pixel intensities of the top row are augmented by pen dynamics, while the bottom row shows binary representations.

5.7 Model Implementation and Training Strategy

The dataset comprises 80 reproductions of Benson Figure Copy drawings. From each drawing, four distinct feature representations were extracted. Due to the limited dataset size, a 5-fold cross-validation scheme was employed to provide robust performance and generalisability estimates. There was only one reproduction per subject, so there was no potential for data leakage.

For the segmentation task, a U-Net architecture was selected, specifically the Dynamic U-Net variant implemented via the fastai deep learning library [73]. The Dynamic U-Net allows for the integration of a pretrained encoder, which is crucial given the limited dataset size. The encoder chosen is a ResNet-34 [65], pretrained on the ImageNet dataset [48]. Pretraining allows the model to leverage prior knowledge; the model has already learned low-level features like edges, shapes and contours, which is particularly appropriate for simple figures such as the Benson Figure. Preliminary experiments showed that utilising this pretrained model led to faster convergence and improved generalisation compared to training from scratch.

The training process leveraged transfer learning through the *fine_tune* method provided by fastai. Initially, all layers except the final ones were frozen, preventing updates to their parameters during the early training stages. This strategy helps retain the beneficial features learned during pretraining while allowing the model to gradually adjust to the new task. The model was first trained using the *fit_one_cycle* method for one epoch, incorporating the 1cycle learning rate policy proposed by Leslie Smith [173], which is known for facilitating more efficient training and better performance.

Following this initial phase, all layers were unfrozen, and *fit_one_cycle* was applied for an additional 10 epochs. This two-stage approach allows the model to progressively adapt to the specific task while minimising the likelihood of forgetting the pretrained features.

A batch size of 2 had to be used due to out of memory errors with larger batch sizes. The GPU used had 10 GB VRAM, which should be capable of larger batch sizes, but this was unable to be resolved.

The Adam optimiser [88] was used with a learning rate of 0.001, $\beta_1 = 0.9$, $\beta_2 = 0.99$, epsilon $= 1 \times 10^{-5}$, and weight decay $= 0.01$. These are the default parameters in fastai and were chosen based on their proven effectiveness in similar deep learning applications [62].

All experiments were conducted on a system equipped with an AMD Ryzen 9 3900 CPU and an Nvidia 3080 GPU with 10GB VRAM and 32GB of RAM.

5.7.1 Post-Processing

The model's output requires further processing to derive meaningful stroke segment labels from pixel-level predictions. To address this, a two-step post-processing methodology is applied.

First, an additional stroke segment ID mask is created for each generated image, where each pixel value corresponds to a unique segment ID. This allows for the aggregation of probability predictions for individual stroke segments. For each stroke segment, the class with the highest mean probability is allocated as the predicted label.

Secondly, a label propagation method accounts for stroke segments not represented in the rasterised drawing. Labels for these missing segments are inferred using a forward fill operation followed by a backward fill, grouped by each stroke. Strokes without any representation are assigned the "UNKOWN" class.

Through this two step approach, the model's pixel level proababilistic outputs are effectively transformed into stroke segment labels.

5.8 Results

The semantic segmentation performance of the U-Net architecture was evaluated on digital reproductions of Benson Figures at two resolutions: 128x128 and 512x512 pixels. Four different representations of pen dynamics were tested: a Base Comparison (constant values in all channels) and three combinations of pen dynamics (Combination 1: x tilt, y tilt, pressure;

Combination 2: x tilt, y tilt, time; Combination 3: pressure, time, constant). Performance metrics were calculated at both the pixel level and the stroke segment level.

5.8.1 Overall Performance

The U-Net architecture demonstrated high performance across all configurations, with accuracies ranging from 93% to 96%. Generally, stroke segment-level metrics were slightly higher than pixel-level metrics, suggesting that the post-processing step effectively improved overall performance.

5.8.2 Performance at 128x128 Resolution

Table 5.2 presents the performance metrics for the 128x128 resolution experiments.

At this resolution, the Base Comparison (constant values in all channels) outperformed all other combinations across all metrics. For pixel-level classification, it achieved an accuracy of 95% ± 4%, precision of 94% ± 5%, recall of 95% ± 4%, F1 score of 93% ± 5%, and Jaccard score of 90% ± 7%. At the stroke segment level, performance was slightly higher, with an accuracy of 96% ± 5%.

Other combinations also performed well, with accuracies ranging from 93% to 94%, but consistently below the Base Comparison.

Figure 5.12 shows the normalised confusion matrices for pixel classification at 128x128 resolution.

Representation	Accuracy	Precision	Recall	F1 score	Jaccard score

Pixel (% ± std)

Base Comparison	**0.95 ± 0.04**	**0.94 ± 0.05**	**0.95 ± 0.04**	**0.93 ± 0.05**	**0.90 ± 0.07**
Combination 1	0.93 ± 0.03	0.93 ± 0.05	0.93 ± 0.03	0.92 ± 0.05	0.88 ± 0.06
Combination 2	0.94 ± 0.04	0.93 ± 0.06	0.94 ± 0.04	0.93 ± 0.05	0.89 ± 0.07
Combination 3	0.93 ± 0.03	0.93 ± 0.05	0.93 ± 0.03	0.92 ± 0.05	0.88 ± 0.06

Stroke segment (% ± std)

Base Comparison	**0.96 ± 0.05**	**0.95 ± 0.06**	**0.96 ± 0.05**	**0.94 ± 0.06**	**0.92 ± 0.08**
Combination 1	0.95 ± 0.04	0.94 ± 0.05	0.95 ± 0.04	0.93 ± 0.06	0.90 ± 0.08
Combination 2	0.95 ± 0.05	0.94 ± 0.07	0.95 ± 0.05	0.94 ± 0.06	0.91 ± 0.09
Combination 3	0.95 ± 0.04	0.93 ± 0.07	0.95 ± 0.04	0.94 ± 0.06	0.90 ± 0.08

Table 5.2: Comparative performance of digital drawing representations for semantic segmentation in digital reproductions of Benson Figures, at a resolution of 128x128. Metrics encompass accuracy, precision, recall, F1 score and Jaccard score. Results presented are the weighted mean of all classes, from 5-fold cross-validation, with the standard deviation also presented. The "Pixel" section pertains to scores derived from the mask generated by the model, while "Stroke segment" metrics relates to scores derived from stroke segment labels recovered back from the mask. The background class is ignored. Strokes with an unknown class are included. Bold values indicate the best performance in the respective metric column.

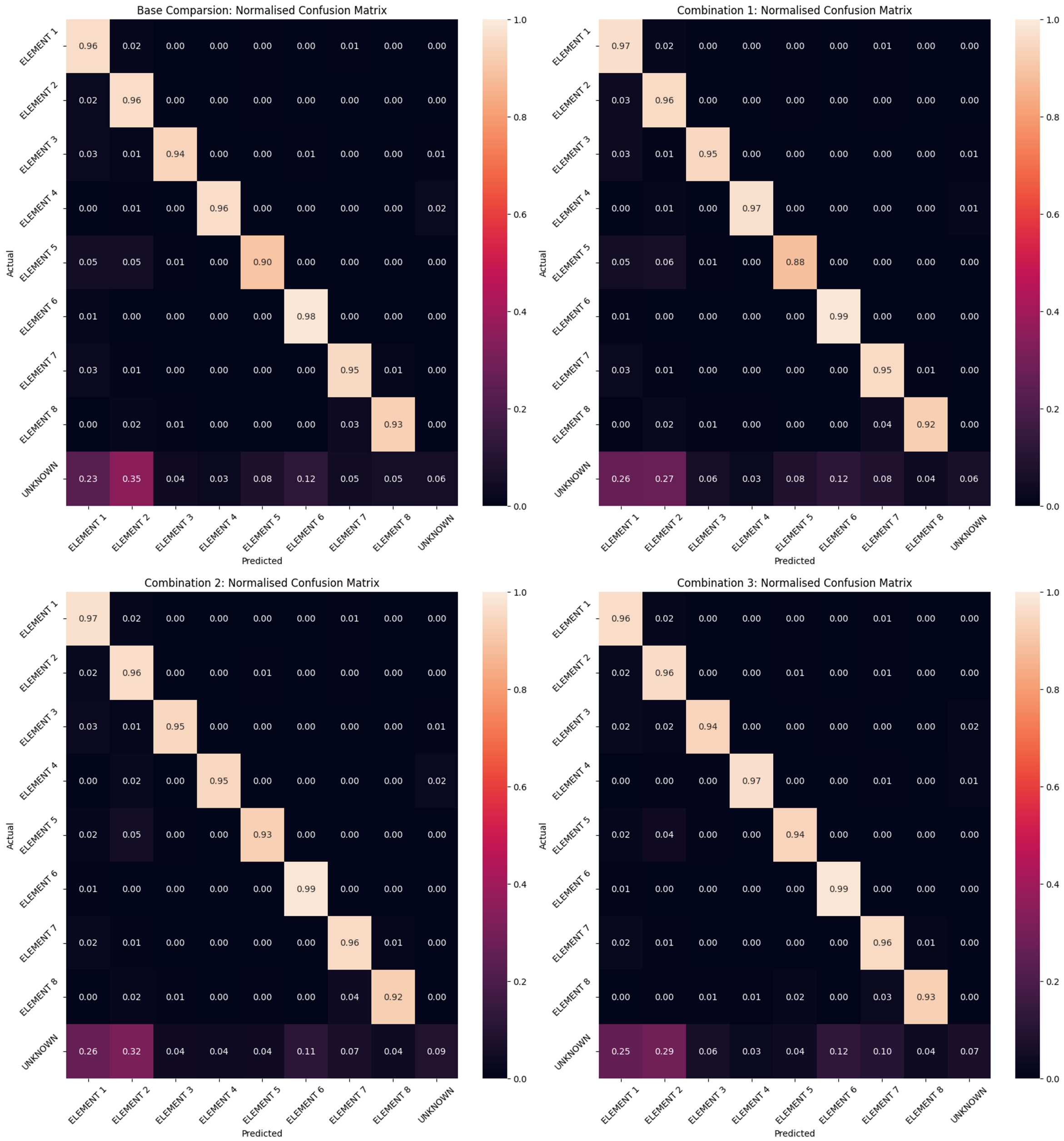


Figure 5.12: Normalised confusion matrices for pixel classification for 128x128 resolution.

5.8.3 Performance at 512x512 Resolution

Table 5.3 presents the performance metrics for the 512x512 resolution experiments. At this higher resolution, Combination 2 (x tilt, y tilt, time) performed best across all metrics. For pixel-level classification, it achieved an accuracy of 94% ± 4%, precision of 93% ± 6%, recall of 94% ± 4%, F1 score of 93% ± 5%, and Jaccard score of 89% ± 7%. Stroke segment-level performance was again slightly higher, with an accuracy of 95% ± 5%.

Other combinations, including the Base Comparison, performed slightly worse but still achieved high scores, with accuracies ranging from 93% to 94%.

Figure 5.13 shows the normalised confusion matrices for pixel classification at 512x512 resolution.

Representation	Accuracy	Precision	Recall	F1 score	Jaccard score
Pixel (% ± std)					
Base Comparison	0.93 ± 0.04	0.92 ± 0.05	0.93 ± 0.04	0.92 ± 0.05	0.88 ± 0.07
Combination 1	0.93 ± 0.04	0.92 ± 0.05	0.93 ± 0.04	0.92 ± 0.05	0.87 ± 0.07
Combination 2	**0.94 ± 0.04**	**0.93 ± 0.06**	**0.94 ± 0.04**	**0.93 ± 0.05**	**0.89 ± 0.07**
Combination 3	0.94 ± 0.04	0.92 ± 0.06	0.94 ± 0.04	0.93 ± 0.05	0.88 ± 0.07
Stroke segment (% ± std)					
Base Comparison	0.95 ± 0.05	0.93 ± 0.07	0.95 ± 0.05	0.94 ± 0.06	0.91 ± 0.08
Combination 1	0.95 ± 0.04	0.94 ± 0.07	0.95 ± 0.04	0.94 ± 0.06	0.91 ± 0.08
Combination 2	**0.95 ± 0.05**	**0.95 ± 0.07**	**0.95 ± 0.05**	**0.94 ± 0.06**	**0.92 ± 0.08**
Combination 3	0.95 ± 0.05	0.94 ± 0.07	0.95 ± 0.05	0.94 ± 0.06	0.91 ± 0.08

Table 5.3: Comparative performance of digital drawing representations for semantic segmentation in digital reproductions of Benson Figures, at a resolution of 512x512. Metrics encompass accuracy, precision, recall, F1 score and Jaccard score. Results presented are the weighted mean of all classes, from 5-fold cross-validation, with the standard deviation also presented. The Pixel section pertains to scores derived from the mask generated by the model, while Stroke segment metrics relate to scores derived from stroke segment labels recovered back from the mask. The background class is ignored. Strokes with an unknown class are included. Bold values indicate the best performance in the respective metric column.

5.8.4 Comparison between resolutions

A notable finding was the difference in optimal representations between resolutions: At 128x128 resolution, the Base Comparison performed best. At 512x512 resolution, Combination 2 (x tilt, y tilt, time) performed best.

Overall, performance metrics were slightly higher for 128x128 resolution compared to 512x512 resolution across all combinations.

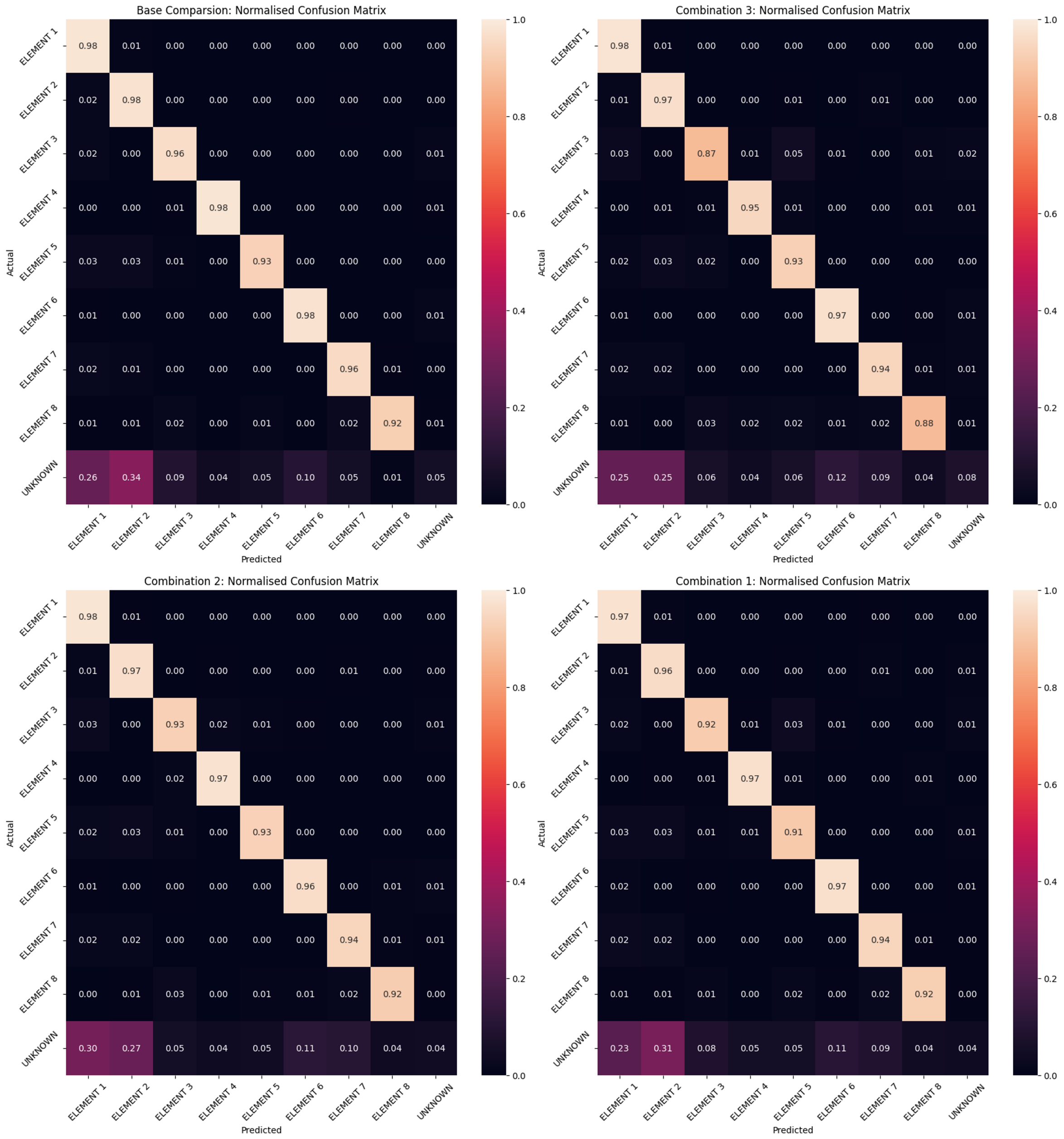


Figure 5.13: Normalised confusion matrices for pixel classification for 512x512 resolution.

The confusion matrices (Figures 5.12 and 5.13) reveal generally good performance across all classes. Notably, the ”UNKNOWN” class appears to have lower performance compared to other classes, most likely due to its rarity in the dataset.

5.9 Discussion

The results demonstrate that the U-Net architecture achieves a high performance in segmenting the Benson Figure, with the best model reaching 96% accuracy, 95% precision, 96% recall, and a 94% F1 score at 128x128 resolution without embedding pen dynamics. This was an unexpected and intriguing finding, as it was expected that the occlusion would be detrimental to the model's performance.

This novel application of deep learning techniques to complex figure segmentation offers promising implications for clinical practice. The model's 96% accuracy in stroke segment labelling means that only about 4% of the labels generated by the model would require correction. This performance is comparable to those of hand-crafted segmentation approaches [24] and [193].

These findings suggests potential significant time savings and reduced cognitive load for clinicans compared to traditional flowchart method and pen-switching method. Additionally, in combination with the applied polyline simplification algorithm applied, this would allow for the seamless extraction of features pertaining to the BQSS [180], and those described in the T-RCF [139]. The overarching aim of this work is to generate a method by which clinicians would be able to extract features that have been highlighted as clinically relevant in the past but were previously to cumbersome to obtain. In this regard, this work also implies that good performance can be acquired will a small dataset, suggesting a potentially rapid improvement in model performance as clinicians engage in the labeling process.

In this regard, this work also implies that good performance can be acquired with a small dataset, suggesting a potentially rapid improvement in model performance as clinicians engage in the labeling process. This could create a positive feedback loop: as more labeled data is acquired through clinical use, the model's performance could improve. In turn, this improved performance could make the labeling process more efficient for clinicians.

However, it's important to highlight that the model struggled with correctly classifying “UNKNOWN” segments, which were rarely present in the dataset. This highlights a potential challenge in generalising to more varied or atypical drawings, such as those produced in recall tasks or by patients with severe cognitive impairments. This limitation could potentially be addressed through the implementation of data augmentation techniques.

Data augmentation is an effective technique for maximising the utility of training data and is commonly employed in computer vision tasks [6]. By generating altered versions of the exisiting samples, the size of the training dataset can be artificially increased. This technique is utilised to prevent overfitting and improve the model's generalisation capabilities, thereby ensuring that the model can effectively handle real-world variability.

For segmentation tasks, data augmentation techniques generally preserve the spatial relationships within the image while introducing variations [6]. Geometric and affine transformations are commonly employed. Translation, rotation, and flipping are used to increase invariance to orientation changes. Scaling is applied to achieve invariance to different object sizes. More advanced techniques include MixUp [205], which combines two images and their corresponding masks by blending them with a ratio, and CutMix [202], which replaces patches of one image with another.

Data-augmentation is particularly applicable in clinical context, where data is often limited. In the works of Park et al., datasets of approximately a couple of hundred images were employed for the segmentation of two other neuropsychological assessments: the Clock-Drawing Test [134] and Interlocking pentagon test [133]. To address this limitation, a U-Net pretrained on the ImageNet dataset was utilised, as was done in this work. Additionally, simple geometric augmentations were applied.

In contrast to previous studies, this research incorporates pen dynamics encoded within the images, introducing additional complexity to the application of data augmentation techniques. For instance, the rotation augmentation would not be accurately reflected in tilt encoding, as a drawing that has been rotated to an angle, would differ significantly from one actually drawn at that angle. For this reason, no data-augmentation was utilised for this work.

Given that encoding pen-dynamics did not demonstrate significant improvement in segmentation performance, numerous opportunities for novel data augmentation techniques specific to rasterised digital drawings remain unexplored. These arise from the fact that the images can be generated with perturbations occurring at any point during the drawing.

Several potential augmentation techniques are proposed. Firstly, varying the order in which strokes are drawn could be considered. This is especially relevant as some stroke segments may not be represented in the final drawing due to overlapping, depending on the resolution, as discussed in 5.6. While this would not alter the input image in a binary representation (the base case), it would change the associated associated stroke segment ID of the pixel in the segmentation mask, and thus potentially the class label. This could potentially aid in model generalisation.

Secondly, the omission of components, strokes and stroke segments could be explored; this would be analogous to the random erasing approach [208]. This study is currently limited by its use of only the Copy portion of the assessment rather than the Recall. While most subjects can reproduce all elements of the figure in the Copy task, this is not guaranteed in the Recall task where subjects may omit some components. As such this augmentation technique could help generalise for these cases.

Thirdly, an expansion of the principles of CutMix to drawings could be implemented. This would involve creating a palette of components drawn by all subjects and generating new images based on this palette.

Finally, to address the lack of representation of UNKNOWN elements in the drawing, strokes that don't relate to any particular element, such as scribbles or lines, could be inserted into a drawing at various points during the task completion.

5.9.1 Limitations

The dataset was labelled by the author, and not a clinician. While the intent during the recall phase is obvious, subjectivity arises when elements of the figure are more vague. This non-clinical labeling could potentially introduce bias or inconsistencies in the dataset. Future work could address this limitation by involving multiple clinicians in the labeling process and assessing inter-rater reliability.

Further studies are needed to validate the impact of these results on clinical efficiency and effectiveness in real-world settings. Such studies should quantify the actual time saved, assess the impact on the clinicians' cognitive load and fatigue, and investigate how this AI assistance affects the overall assessment and diagnosis process. Additionally, it would be valuable to explore how the model performs with a more diverse set of drawings, including those from patients with various levels of cognitive impairment.

5.10 Conclusion

This chapter has demonstrated the successful application of deep learning techniques to the automated segmentation of Benson Figure, a simplified complex figure used in neuropsychological assessments. The U-Net architecture achieved high accuracy in segmenting the figure's elements, demonstrating potential for significant time savings in clinical practice. While the novel encoding of pen dynamics did not substantially improve segmentation performance as initially hypothesized, this approach could potentially be more applicable to other computer vision applications, such as PD diagnosis from graphmotor assessments. The creation of a bespoke annotation tool establishes a foundation for extending this work to more complex figures like the Rey-Osterrieth Complex Figure. These results indicate promising directions for enhancing the efficiency and objectivity of neuropsychological assessments. However, further research is needed to validate the clinical impact of these findings, address current limitations, and explore applications in diverse patient populations with varying levels of cognitive impairment.

Chapter 6

Summary and Conclusions

This thesis has examined a range machine learning methodologies aimed at the automatic, objective and quantitative assessment of Parkinson's disease. A substantial portion of this work is dedicated to exploring innovative techniques designed to enhance the utility of digitsed standard clinical assessments. This concluding chapter offers a summary of the key findings and novel contributions made in each respective chapter. Additionally, it revisits the hypothesis and research aims stated in Chapter 1.

6.1 End-to-End learning for the MDS-UPDRS Part III Assessments

6.1.1 Rationale and Work Conducted

The capabilities of computer vision and machine learning in evaluating the hand kinematics, has been gaining interest. Traditional approaches have relied on feature engineering that reduces high-dimensional data, potentially overlooking critical diagnostic features. Newer techniques, such as deep learning and end-to-end multivariate time series algorithms, offer automated feature extraction and quick training times, making them advantageous for broader applications in medical diagnosis and severity assessment of motor conditions.

6.1.2 Novelty and Contribution

This study shows the first application of state of the art multivariate timeseries classification models, InceptionTime, MiniRocket and MultiRocket, to clinically standard upper limb assessment tasks.

6.1.3 Key Findings

1. Using the study conducted by Williams et al.[197], it was found that the models trained in this study performed exceeded the accuracy of neurologists ability to differentiate between healthy controls and PD from finger tapping recordings.

2. The algorithms were found to be effective in developing classifiers for all three bradykinesia tasks outlined in the MDS-UPDRS part III scale. The pronation-supination task was found to be the most discriminating task for both PD vs. healthy control and bradykinesia vs. no bradykinesia classification.

3. The ability of these algorithms to accept raw positional data facilitates an unbiased search of the solution space.

4. Misclassification analysis indicated expectedly, slight severity scores were more likely to result in false positives, as is seen in clinical experts.

6.1.3.1 Limitations

A significant and unfortunate limitation of this study was the necessary trimming/windowing of recordings time to 5 seconds, although the results of the trained models are very promising for future research, this is half the time required for the MBRS and half the time. Only one neurologist the generalising capabilities and robustness of the automated assessment is limited.

6.1.4 Practical Implications

The rising prominence of wearable devices in healthcare underscores the need for accurate timeseries models. Although expert domain knowledge is ideal for model development, the growing number of applications and pace of technological advancements necessitate automated approaches. This experiment substantiates that time-series classification models can provide accurate classification comparable to those requiring expert knowledge and preprocessing time.

6.1.4.1 Future Directions

Given these findings, future work may focus on clincial assessments that have not be quantified digitally, such as the Luria motor sequence, can be initially explored using this approach.

6.2 Semantic Segmentation of Neuropsychological Figures

6.2.1 Rationale and Work Conducted

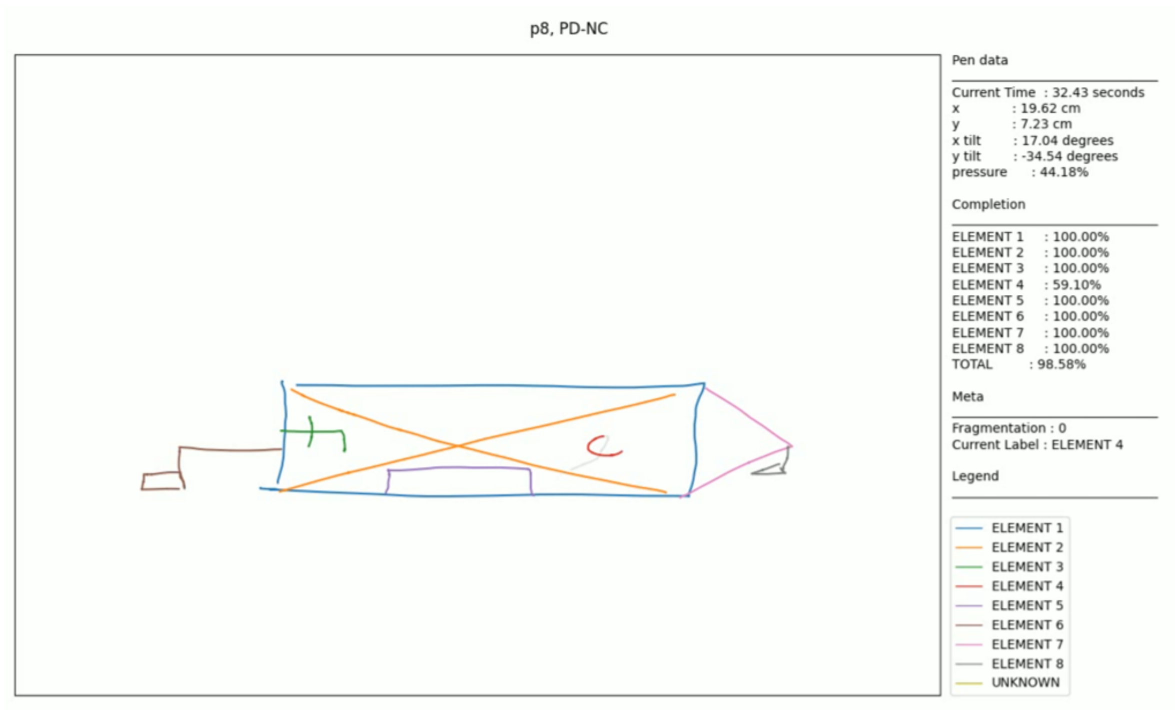
The study aimed to mitigate the inherent limitations of existing neuropsychological assessments that rely on manual methods for evaluating tasks like the ROCF. By employing digital graphics tablets, 80 Benson Figure reproductions were collected from Parkinson's Disease patients and age-matched healthy controls at Leeds Hospital. A software platform for manual labelling was developed, featuring the Ramer-Douglas-Peucker (RDP) algorithm for sophisticated stroke segmentation. A U-net model with a pretrained ResNet34 encoder was trained to perform automated segmentation of these complex figures. Multiple experiments were conducted to evaluate the effects of different resolutions and pen dynamic encoding.

6.2.2 Key Findings

The research yielded several important findings. Firstly, a binary encoding representation resulted in the highest segmentation accuracy, at 96%. Secondly, adjustments to resolution levels had minimal impact on stroke segment accuracy. Lastly, varying encoding parameters related to pen dynamics did not significantly alter the segmentation performance compared to the base representation.

6.2.3 Novelty and Contribution

This research presents an innovative approach for semantic segmentation of complex neuropsychological figures, thereby alleviating the manual burden of stroke segmentation. This



```
Pen data

Current Time  : 32.43 seconds
x             : 19.62 cm
y             : 7.23 cm
x tilt        : 17.04 degrees
y tilt        : -34.54 degrees
pressure      : 44.18%

Completion

ELEMENT 1    : 100.00%
ELEMENT 2    : 100.00%
ELEMENT 3    : 100.00%
ELEMENT 4    : 59.10%
ELEMENT 5    : 100.00%
ELEMENT 6    : 100.00%
ELEMENT 7    : 100.00%
ELEMENT 8    : 100.00%
TOTAL        : 98.58%

Meta

Fragmentation : 0
Current Label : ELEMENT 4
```

Figure 6.1: Quantifying organisational strategy.

automated process has the potential to capture more nuanced data in assessments, enhancing diagnostic precision and patient management.

6.2.3.1 Limitations

The labelling of the dataset was conducted by the author. The RDP method will be slower than labelling strokes entire strokes and adding manual breakpoints to begin with.

6.2.4 Practical Implications

The automated labelling method demonstrated here is feasible even with a small dataset. It also allows for a positive feedback loop, as a neurologist manually labels data, the model can be incrementally retrained to enhance its performance.

6.2.4.1 Future Directions

Initial investigations into the calculation of organizational features such as fragmentation have begun, laying groundwork for further research in automated, nuanced neuropsychological assessments.

6.3 Overall Conclusions

Revising the hypothesis that,

Machine learning methodologies can serve as effective tools in improving diagnostic utility of standard clinical assessments in Parkinson's Disease

Given that an effective classifier has been trained to diagnose PD, with a greater accuracy than that of clinicians, and a efficient approach for semantic labelling of psychological figures has been proposed, we can conclude that they can.

Appendix A

Algorithms

Algorithm 1: Ramer-Douglas-Pueker Algorithm for Polyline Simplification

```
function RDP ($P, \epsilon$);
Input  : List of points $P = [p_0, p_1, \ldots, p_n]$, tolerance $\epsilon$
Output: Reduced list of points
// Find the point with the maximum distance to the line segment
   composed of $p_0$ and $p_n$
$d_{max} \leftarrow 0$;
$index \leftarrow 0$;
for $i = 1$ to $n - 1$ do
    $d \leftarrow$ PerpendicularDistance($p_i, p_0, p_n$);
    if $d > d_{max}$ then
        $index \leftarrow i$;
        $d_{\mathrm{max}} \leftarrow d$;
    end
end
// If max distance is greater than epsilon, then recursively simplify
if $d_{max} \geq \epsilon$ then
    $results_1 \leftarrow$ RDP($P[0 : index], \epsilon$);
    $results_2 \leftarrow$ RDP($P[index : n], \epsilon$);
    return concatenate($results_1, results_2$)
else
    return $[p_0, p_n]$
end
```

Appendix B

Application Development

B.1 Introduction

Data acquisition and labelling are foundational tasks in every supervised machine learning study, providing the datasets needed for model training and evaluation. Throughout this thesis, various tools were developed to facilitate these tasks. Acquisition tools were designed to guide assessors through clinical protocols, capturing data from a diverse range of sensors, including graphics tablets, EM positional tracking sensors, and eye-tracking glasses. These were complemented by the development of annotation tools for labelling the collected time-series data.

A key observation made during this process is that existing tools often fail to meet the specific requirements of individual studies. Consequently, the ability to create adaptable, specialised tools may prove invaluable for researchers. This appendix chapter highlights the design and implementation of an annotation tool using Ignition, an integrated software platform for Supervisory Control and Data Acquisition (SCADA) systems. SCADA systems are widely utilised in modern society, ranging from industrial automation to national infrastructure, to monitor and control physical processes through graphical interfaces. Despite its primary use in industrial contexts, Ignition's server-centric web deployment model and user-friendly application development are worth highlighting, as they may prove relevant in further studies that need to develop bespoke tooling.

B.2 Motivation

The motivation for this study, as presented in Chapter 4, involves the analysis of three hand kinematic assessments utilised in the MDS-UPDRS to evaluate motor impairment. These assessments were recorded using an glsem tracking system, which captured the position and orientation of the index finger and thumb during each task. The three assessments - Finger-Tapping, Pronation-Supination, and Hand Open-Close, are inherently periodic tasks and the frequencies of the movements were investigated as part of this work. From each recording a uni-variate signal was used as a proxy for the movement: Euclidean distance between the finger and thumb for Finger Tapping, and Hand Open-Close, and roll of the index finger for Pronation-Supination.

The dataset utilised in this thesis comprises recordings from Parkinson's patients and healthy controls. These recordings could serve as the basis for training feature-based machine learning models aimed to differentiate between PwPD and healthy individuals. One such model employed by [114] is a Random Forest classifier, reliant on features that quantify the signal. Extraction of these features necessitates segmentation of the signal into phases, achieved through a peak detection algorithm. Tuning the parameters of this algorithm requires manually labelling the peaks in each signal to calculate the error between the ground truth and algorithm-detected peaks. Thus, there is a clear need for specialised software to facilitate this manual labeling process.

B.3 Requirements

A distinction must be made between the recording file generated by the measurement system and the metadata about the recording itself, as each has separate annotation requirements. For the recording file, individual samples may possess labels such as "start", "end", "anomaly", "peak", or "trough", either singularly or in combination. In contrast, a record might include annotations such as "Exclude: Contains hemisphere switching" (an error observed during the study) or "Exclude: No usable data". From these distinctions, the following requirements can be derived:

1. **Separate views for database records and their contents**: The application must provide distinct interfaces for viewing and annotating both the overall records and the individual data samples within each record.

2. **Multi-label support**: Both records and samples should support multiple simultaneous labels to accommodate complex annotation scenarios.

3. **Efficient navigation**: Given the large dataset (1567 files total, including 395 calibration files, 391 Finger Tapping files, 391 Hand Open-Close files, and 390 Pronation-Supination files), the application must provide an efficient means for users to navigate through the files in the database.

4. **Flexible data visualization**: Different tasks yield different signals representing motion (Euclidean distance for finger tapping, hand-open close, and calibration; roll of the index finger for Pronation-Supination). Therefore, users should be able to select which channels from the raw multivariate time-series data are displayed at any given time.

5. **Adaptability**: The application should be flexible enough to accommodate different types of kinematic assessments and annotation schemes, allowing for its use in various research contexts.

6. **Collaborative features**: Given the volume of data to be annotated, the system should support multiple users working concurrently, with appropriate mechanisms to track progress and prevent conflicts.

B.4 Evaluation of Requirements Against Available Open-Source Tools

Three applications were identified as potential solutions for this task:

1. **Curve, Baidu** [9]: is a JavaScript-based web application. It uses a three-column Comma-Separated Value (CSV) file format where the first column represents the timestamp, the second the value, and the third the label. The labeling is binary, with 0 for normal and 1 for abnormal. Curve lacks the ability to annotate records, support multiple labels, or handle multivariate time-series data.

2. **TagAnomaly, Microsoft** [112]: is an R-based project that, similar to Curve, deals with univariate series. It does offer the option to assign more than two categories, but only one category per sample is allowed. Files are in CSV format and must be loaded individually. TagAnomaly also lacks support for record-level annotations and integrated dataset navigation.

3. **The Wearables Development Toolkit, Technical University of Munich** [63]: offers the most functionality among the three. This MATLAB-based application can navigate between files (.mat format) in a dataset, and labels can be predefined. Users can annotate events that occur at a specific moment in time or activities with a duration. The app also has the functionality to load and display videos alongside the data. Additionally, multiple channels of a multivariate signal can be displayed simultaneously. However, records themselves cannot be annotated, navigation between records is not efficient and samples only support one annotation at a time.

While certain useful features are offered by each of these tools, the specific requirements of this annotation task are not fully met by any of them. The primary limitations identified across these tools include the lack of support for both record-level and sample-level annotations, limited or no capability for multiple simultaneous labels, insufficient flexibility in handling diverse data types and annotation schemes, and the absence of collaborative features for multi-user annotation.

The need for a custom solution is underscored by these limitations. A bespoke tool, as described in subsequent sections, is aimed at overcoming these limitations.

B.5 Design

The architecture consists of a single Ignition server instance (also known as the Gateway) connected to an instance of MariaDB a relational database management system. With the Gateway, multiple clients can access the application using devices (including movile devices) that support a modern web browser. A visualisation of the diagram is given in Figure B.1.

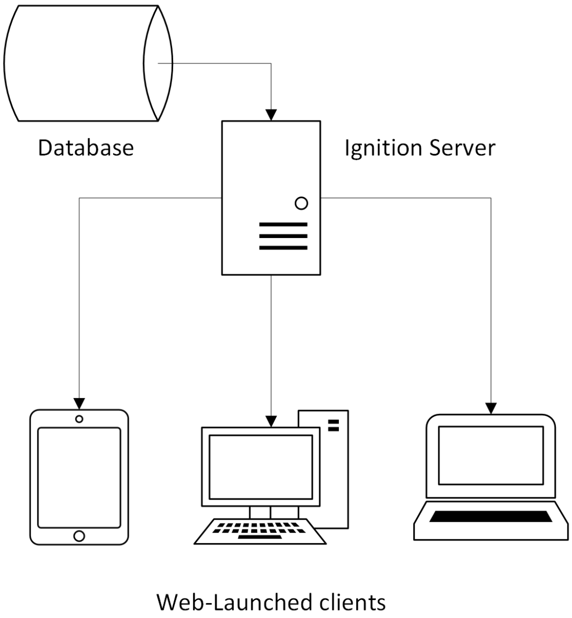


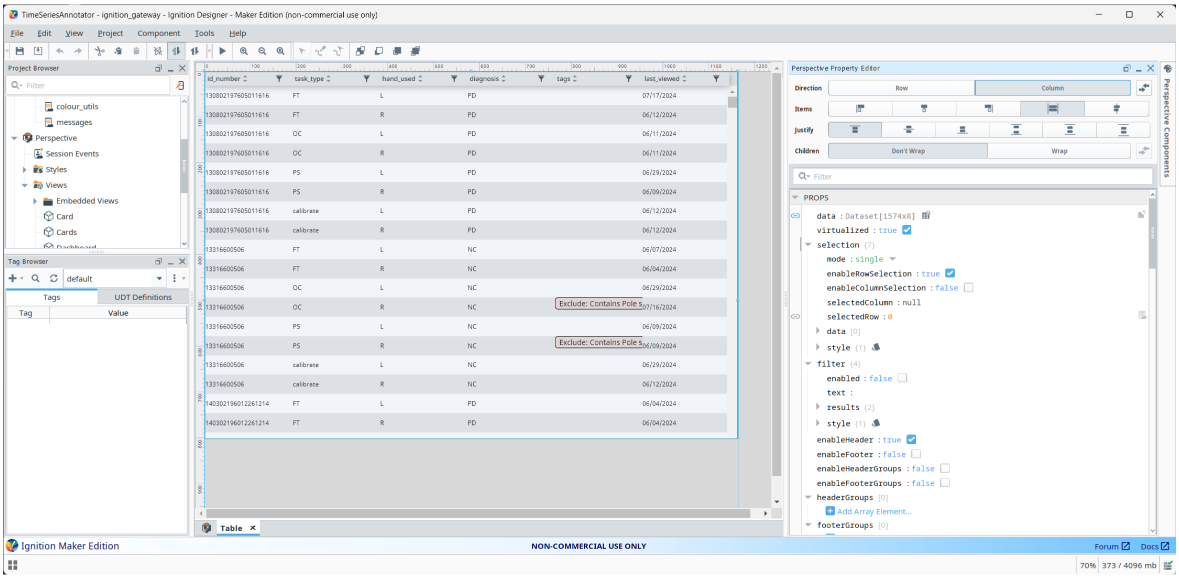
Figure B.1: A visualisation of the schema used in the application.

B.5.1 Ignition

The selection of Ignition for this project was motivated by its comprehensive solution for application development and deployment, a feature that otherwise from the ground up would necessitate a high proficiency in software development and substantial time investment. This aspect is especially advantageous in research environments where time and resources are often constrained, and the primary focus should remain on the research itself rather than extensive software development. Additionally, the software is free, Ignition has a non-commercial licence that retains the functionality of the full commerical product.

B.5.1.1 Advantages in Development

The Ignition Designer serves as the primary interface for configuration and design work. It is "low-code" integrated development environment, applications are built visually through a drag-and-drop interface, the designer interface is shown in in Figure B.2. It is crucial to note, however, that this does not mean that it is "no-code". Ignition provides users with Python scripting capabilities, enabling the implementation of complex functionality when required [1]. An example of scripting is shown in Figure B.3. This dual approach of visual design and scripting capabilities allows for a balance between accessibility and advanced functionality, catering to the diverse skill sets often found in interdisciplinary teams.



id_number	task_type	hand_used	diagnosis	tags	last_viewed
130802197605011616	FT	L	PD		07/17/2024
130802197605011616	FT	R	PD		06/12/2024
130802197605011616	OC	L	PD		06/11/2024
130802197605011616	OC	R	PD		06/11/2024
130802197605011616	PS	L	PD		06/29/2024
130802197605011616	PS	R	PD		06/09/2024
130802197605011616	calibrate	L	PD		06/12/2024
130802197605011616	calibrate	R	PD		06/12/2024
13316600506	FT	L	NC		06/07/2024
13316600506	FT	R	NC		06/04/2024
13316600506	OC	L	NC		06/29/2024
13316600506	OC	R	NC	Exclude: Contains Pole s	07/16/2024
13316600506	PS	L	NC		06/09/2024
13316600506	PS	R	NC	Exclude: Contains Pole s	06/09/2024
13316600506	calibrate	L	NC		06/29/2024
13316600506	calibrate	R	NC		06/12/2024
140302196012261214	FT	L	PD		06/04/2024
140302196012261214	FT	R	PD		06/04/2024

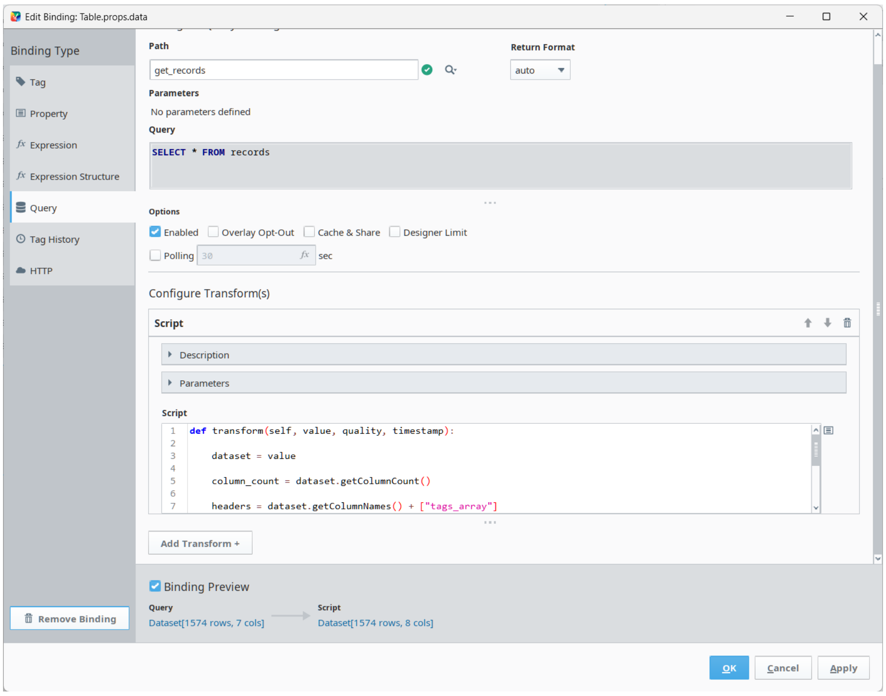
```
data : Dataset[1574x8]
virtualized : true ☑
selection {7}
  mode : single
  enableRowSelection : true ☑
  enableColumnSelection : false ☐
  selectedColumn : null
  selectedRow : 0
  data [0]
  style {1}
filter {4}
  enabled : false ☐
  text :
  results {2}
  style {1}
enableHeader : true ☑
enableFooter : false ☐
enableHeaderGroups : false ☐
enableFooterGroups : false ☐
headerGroups [0]
  Add Array Element...
footerGroups [0]
```

Figure B.2: The Ignition Designer application. The image shows the view with the configured table component for the application. This is a reusable view that can be embedded elsewhere in the application.

[1] Ignition utilises the Jython interpreter (as opposed to CPython, which is synonymous with Python) to execute Python code. Consequently, many standard Python libraries are not compatible, however there is a community driven exchange available for sharing resources.

A key component of Ignition is the Perspective module, which leverages JavaScript, HTML, and CSS to create responsive and interactive user interfaces for modern browsers. Perspective promotes object-oriented design patterns through its use of views, which are reusable components that can be nested and parameterised. This approach encourages modular design and reuse of created views, which can significantly speed up development and improve maintainability. Additionally, the Designer also includes a built-in view testing feature, allowing developers to preview and interact with their designs in real-time without needing to deploy the project. This immediate feedback loop is invaluable for rapid prototyping and iterative development.

Furthermore, since Ignition is hosted on a server, projects can be worked on simultaneously by multiple developers. This feature could facilitate collaboration between technical and non-technical team members, allowing researchers with different skill sets to contribute effectively to the project. For instance, a domain expert could focus on the logical flow and data representation, while a more technically inclined team member could handle complex scripting tasks.



Edit Binding: Table.props.data

Path: get_records

Return Format: auto

Parameters

No parameters defined

Query

```
SELECT * FROM records
```

Options

☑ Enabled □ Overlay Opt-Out □ Cache & Share □ Designer Limit

□ Polling 30 sec

Configure Transform(s)

Script

```
def transform(self, value, quality, timestamp):

    dataset = value

    column_count = dataset.getColumnCount()

    headers = dataset.getColumnNames() + ["tags_array"]
```

☑ Binding Preview

Query: Dataset[1574 rows, 7 cols] → Script: Dataset[1574 rows, 8 cols]

Figure B.3: An example of scripting in ignition. This is an example of a transformation binding on the data property of the table displaying all of the records in the database. A python script is used to transform the output of the query to add an additional column, that contains an array of decoded labels for the record.

B.5.1.2 Advantages in Deployment

The Ignition gateway handles many of the complex back-end operations. These include:

- **Security**: Critical features include user authentication, data encryption (SSL/TLS encryption for all network communications), audit trails, and session management. These are particularly important in biomedical research contexts where data privacy and integrity are paramount, and for complying with GDPR regulations.

- **Client-Server architecture**: The Ignition Gateway acts as a central server, clients access the application through web browsers, without needing to install additional software. The Gateway can serve multiple multiple clients simultaneously from different locations. Each client will have access to the same data, as databases are only connected to the Gateway.

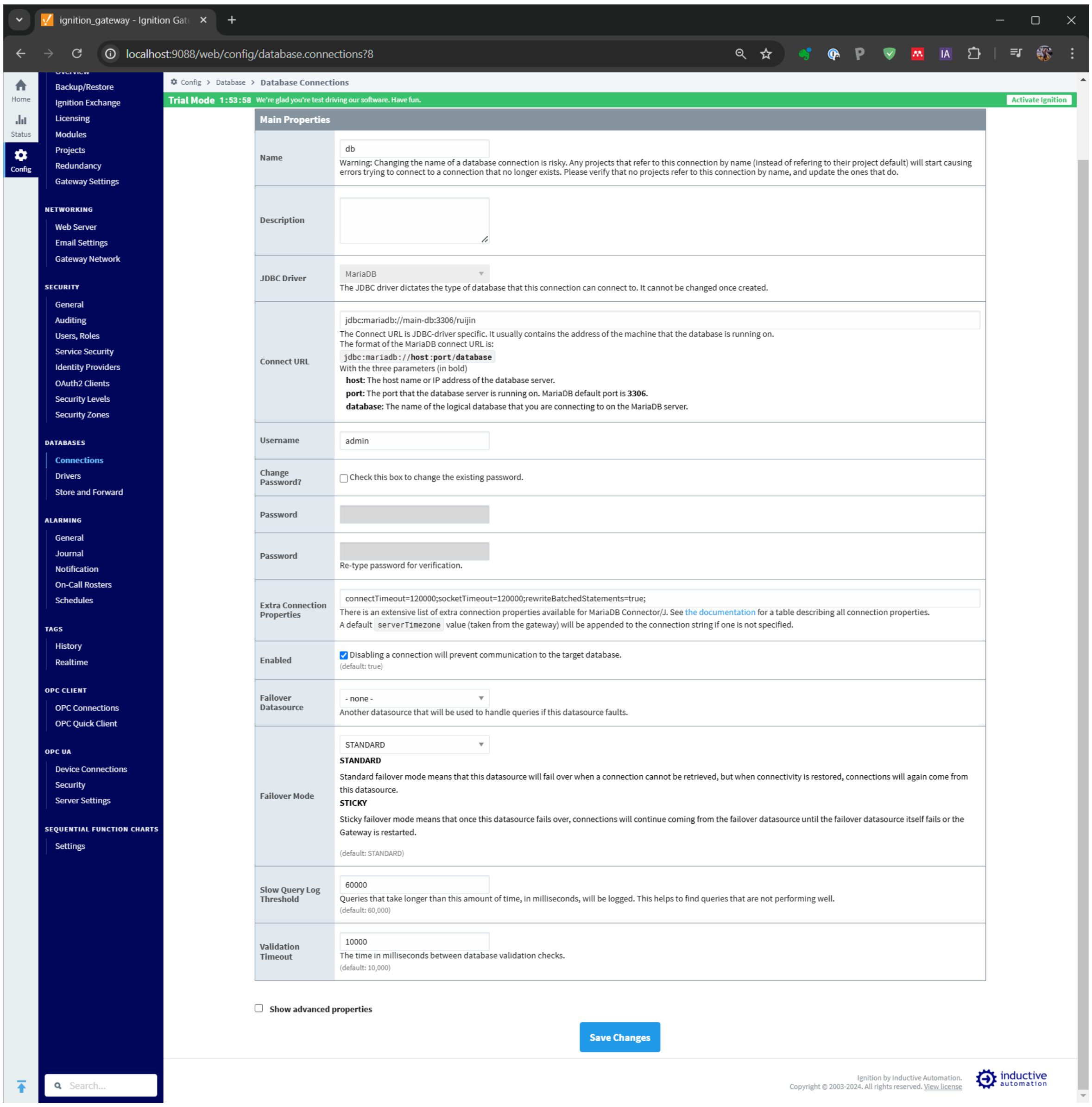
- **Data Connectivity**: Ignition can communicate with and integrate data from multiple diverse data sources. Ignition can connect to various types of databases (SQL, MySQL, Oracle, etc., and example of this configuration is shown in Figure B.5), web services and other systems using REST APIs and IoT devices using the MQTT protocol [149].

B.5.2 Database

The development of the application necessitated a choice in a data storage solution. While traditional file-based storage methods like CSV files are commonly used in research contexts due to their simplicity and portability, a relational database system was selected for this project. This decision was driven by several key factors that aligned with the requirements of the annotation process.

The multi-dimensional nature of the data, encompassing time-series sensor readings, participant information, and various annotation tags, naturally aligns with a structured, relational model. Unlike flat file structures, a relational database enables efficient organisation of this data into logical tables with defined relationships, facilitating more intuitive data management and retrieval. The annotation process involves potential concurrent work by multiple users on the same dataset. Relational databases excel in handling concurrent access, ensuring data consistency and preventing conflicts that could arise in file-based systems. Additionally, the use of SQL provides a standardised, powerful method for data manipulation and retrieval. Furthermore, the choice of a relational database system ensures compatibility with Ignition.

In summary, while file-based storage methods have their merits, the selection of a relational database for this project was driven by the need for structured data organisation, support for concurrent access, powerful query capabilities, and compatibility with the chosen development platform.



Main Properties

Name	db Warning: Changing the name of a database connection is risky. Any projects that refer to this connection by name (instead of refering to their project default) will start causing errors trying to connect to a connection that no longer exists. Please verify that no projects refer to this connection by name, and update the ones that do.
Description	
JDBC Driver	MariaDB The JDBC driver dictates the type of database that this connection can connect to. It cannot be changed once created.
Connect URL	jdbc:mariadb://main-db:3306/ruijin The Connect URL is JDBC-driver specific. It usually contains the address of the machine that the database is running on. The format of the MariaDB connect URL is: `jdbc:mariadb://host:port/database` With the three parameters (in bold) **host:** The host name or IP address of the database server. **port:** The port that the database server is running on. MariaDB default port is **3306**. **database:** The name of the logical database that you are connecting to on the MariaDB server.
Username	admin
Change Password?	☐ Check this box to change the existing password.
Password	
Password	Re-type password for verification.
Extra Connection Properties	connectTimeout=120000;socketTimeout=120000;rewriteBatchedStatements=true; There is an extensive list of extra connection properties available for MariaDB Connector/J. See the documentation for a table describing all connection properties. A default `serverTimezone` value (taken from the gateway) will be appended to the connection string if one is not specified.
Enabled	☑ Disabling a connection will prevent communication to the target database. (default: true)
Failover Datasource	- none - Another datasource that will be used to handle queries if this datasource faults.
Failover Mode	STANDARD **STANDARD** Standard failover mode means that this datasource will fail over when a connection cannot be retrieved, but when connectivity is restored, connections will again come from this datasource. **STICKY** Sticky failover mode means that once this datasource fails over, connections will continue coming from the failover datasource until the failover datasource itself fails or the Gateway is restarted. (default: STANDARD)
Slow Query Log Threshold	60000 Queries that take longer than this amount of time, in milliseconds, will be logged. This helps to find queries that are not performing well. (default: 60,000)
Validation Timeout	10000 The time in milliseconds between database validation checks. (default: 10,000)

☐ Show advanced properties

Save Changes
Ignition by Inductive Automation.
Copyright © 2003-2024. All rights reserved. View license
inductive automation

Figure B.4: A screenshot of the current database configuration for the database used in the application. This is accessed through the Ignition Gateway's web client.

B.5.2.1 Schema

A core principle in the design of the application is that it should be adaptable to different datasets. Consequently, the application dynamically adjusts labelling options, table columns, and charted signals based on the database table contents. The database schema (illustrated in Figure B.5) consists of the following structure and features:

- **Core Annotation Tables:** The record_tags and sample_tags tables store all possible tags for records and individual data samples. The 'id' field serves a dual purpose: it acts as a primary key and determines the tag's position in the one-hot encoded representation. For instance, a tag with id=2 would be represented by the bit '100' in the encoded integer. The 'tag' field contains the human-readable description of the annotation.

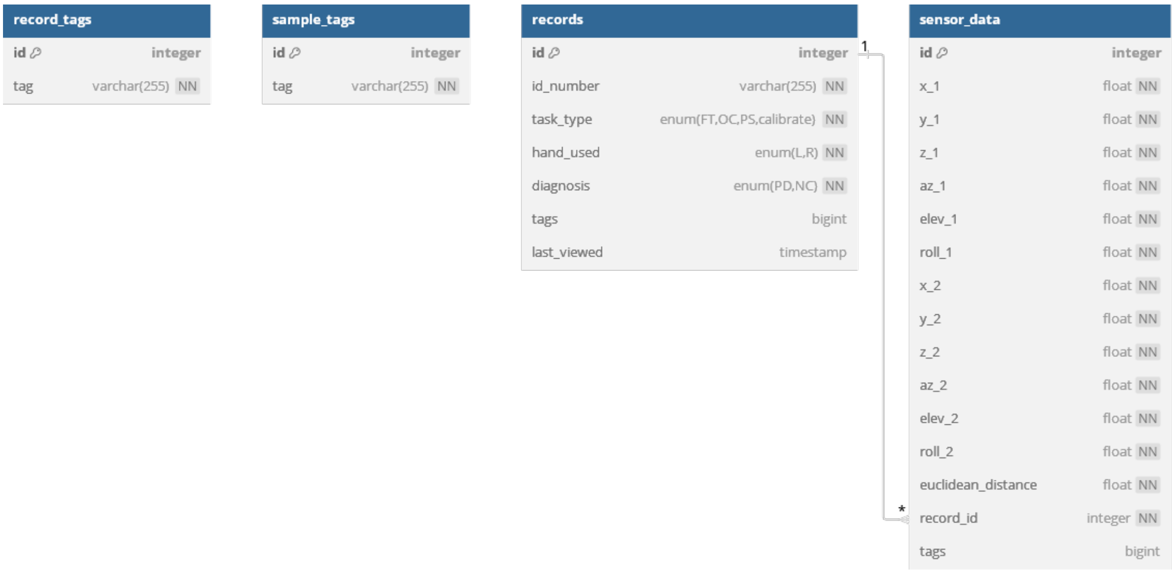
- **Records:** The records table contains metadata for each recording session. The required fields for this table are 'id', 'tags', and 'last_viewed'. The 'id' field uniquely identifies each record, the 'tags' field employs a 32-bit integer for one-hot encoding of record-level annotations, and 'last_viewed' tracks user interactions with the record via the application. Additional fields such as 'id_number', 'task_type', 'hand_used', and 'diagnosis', are all specific metadata associated with this dataset's recordings. These columns facilitate filtering functionality, and decision making processes. For example, depending on the 'task_type', the user would select different channels from the sensor data to be displayed.

- **Sensor Data Table:** This table stores the actual time-series data, each row represents a single sensor data sample. All sensor data is consolidated in this table, the 'record_id' column is used to group all samples from the same recording.

- **Annotation Mechanism:** Each element in the records or sensor_data table needs the capability to have one or more labels. Given MariaDB's lack of an array datatype, a one-hot encoding system is implemented. For example, if a record has tags with ids 1, 3, and 5, its 'tags' field would contain the binary value 10101 (decimal 21). This system allows for up to 32 different tags per record or sample.

- **Schema Flexibility:** While the current implementation includes specific fields in the records and sensor_data tables, the schema is designed for adaptability. The records table can be modified to include different metadata fields relevant to the specific study. Similarly, the sensor_data table can be adjusted to accommodate various types and numbers of sensor channels, with only the 'id', 'record_id', and 'tags' fields being mandatory.

- **Integration with the Application:** The application dynamically reads the record_tags and sample_tags tables to populate the available annotation options. When a user applies a tag, the corresponding bit in the 'tags' field is set to 1. This design allows the application to adapt to different annotation schemes without requiring code changes.



record_tags		
id	integer	
tag	varchar(255)	NN

sample_tags		
id	integer	
tag	varchar(255)	NN

records		
id	integer	
id_number	varchar(255)	NN
task_type	enum(FT,OC,PS,calibrate)	NN
hand_used	enum(L,R)	NN
diagnosis	enum(PD,NC)	NN
tags	bigint	
last_viewed	timestamp	

sensor_data		
id	integer	
x_1	float	NN
y_1	float	NN
z_1	float	NN
az_1	float	NN
elev_1	float	NN
roll_1	float	NN
x_2	float	NN
y_2	float	NN
z_2	float	NN
az_2	float	NN
elev_2	float	NN
roll_2	float	NN
euclidean_distance	float	NN
record_id	integer	NN
tags	bigint	

Figure B.5: A visualisation of the schema used in the application. Created using dbdiagram.io [71]

B.6 Implementation

B.6.1 Application

The application as illustrated in Figure B.6 comprises five embedded views:

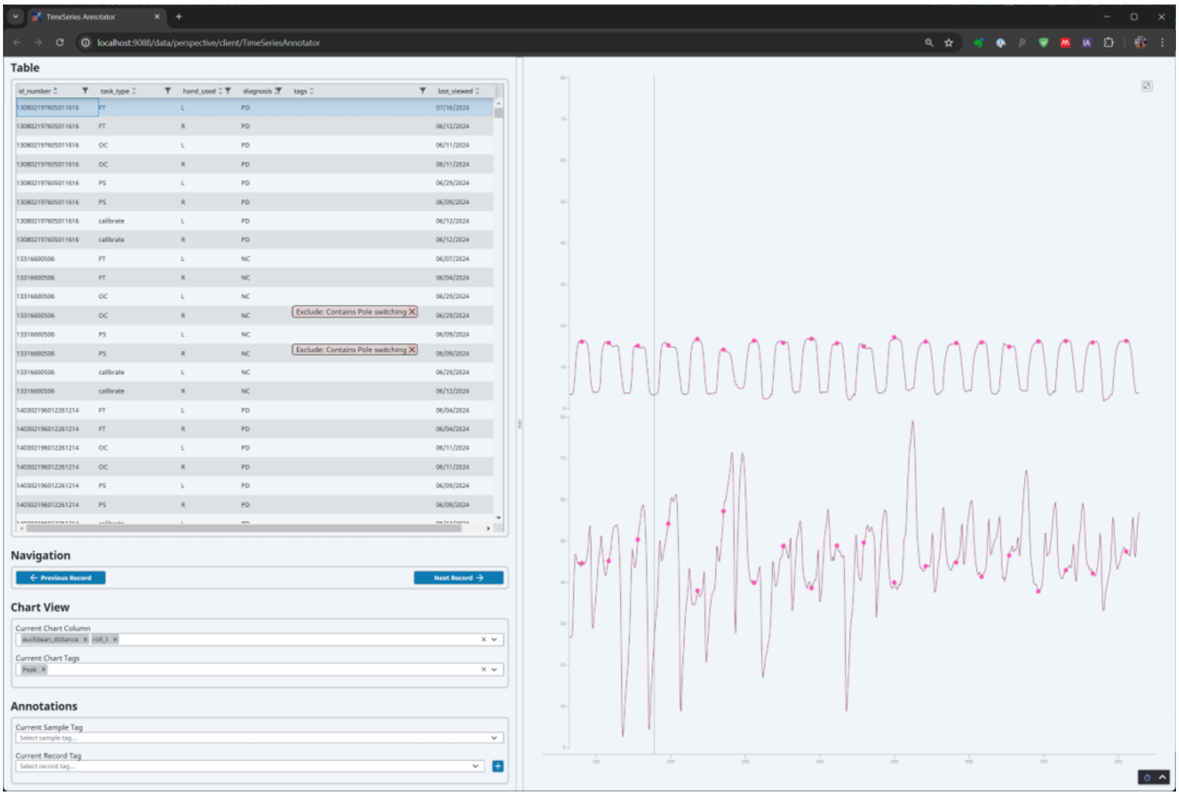
1. **Table View**: This component displays a table populated with data from the records_table. A key feature is the interactive tag component, which decodes and presents the tags for each record. Each tag is accompanied by a removable 'x' icon; when activated, it triggers an update query to clear the corresponding tag's bit. The table refreshes upon each update, dynamically reflecting changes in tag visibility. Rows are selectable, with selection prompting the display of the associated recording on the chart. Additionally, each column supports sorting and filtering functionalities (Figure B.7).

2. **Navigation View**: This view incorporates two buttons for incrementing or decrementing the selected row id in the table. Scripting is employed to ensure the resulting row id remains within the table's bounds.

3. **Chart View**: Two drop-down list components in this view facilitate the customization of the time-series chart display. The "current chart columns" selector allows users to choose which channels from the sensor_data table are visualised. Each option can be toggled, with the time-series chart dynamically adjusting its subplots to reflect the current selection. Similarly, the "current chart tags" selector, populated from the sensor_tags table, controls the display of current annotations. Samples with tags corresponding to the selected options are denoted by circles on the chart, with each unique sample tag automatically assigned a distinct color.

4. **Annotations View**: This view presents two drop-down lists for selecting the user's current annotations for records and samples. These lists are dynamically populated from queries to the record_tags and sample_tags tables. Updates to the currently selected record can be executed from this view via an adjacent plus button, while sample updates are made directly from the time-series chart view.

5. **Time-Series Chart View**: Occupying the right half of the application, this view presents the time-series data. The displayed channels and annotations are controlled via the aforementioned Chart View drop-down lists. Users can interact with the chart through mouse wheel operations for zooming and panning. The current sample, indicated by an X-trace (current position of the mouse on the chart), can have its annotation set or cleared using the 'a' and 'd' keys respectively.



Table

id_number	task_type	hand_used	diagnosis	tags	last_viewed
130802197605011616	FT	L	PD		07/16/2024
130802197605011616	FT	R	PD		06/12/2024
130802197605011616	OC	L	PD		06/11/2024
130802197605011616	OC	R	PD		06/11/2024
130802197605011616	PS	L	PD		06/29/2024
130802197605011616	PS	R	PD		06/09/2024
130802197605011616	calibrate	L	PD		06/12/2024
130802197605011616	calibrate	R	PD		06/12/2024
13316600506	FT	L	NC		06/07/2024
13316600506	FT	R	NC		06/04/2024
13316600506	OC	L	NC		06/29/2024
13316600506	OC	R	NC	Exclude: Contains Pole switching ✕	06/29/2024
13316600506	PS	L	NC		06/09/2024
13316600506	PS	R	NC	Exclude: Contains Pole switching ✕	06/09/2024
13316600506	calibrate	L	NC		06/29/2024
13316600506	calibrate	R	NC		06/12/2024
140302196012261214	FT	L	PD		06/04/2024
140302196012261214	FT	R	PD		06/04/2024
140302196012261214	OC	L	PD		06/11/2024
140302196012261214	OC	R	PD		06/11/2024
140302196012261214	PS	L	PD		06/09/2024
140302196012261214	PS	R	PD		06/09/2024

Navigation

Chart View

Current Chart Column: euclidean_distance ✕ roll_1 ✕

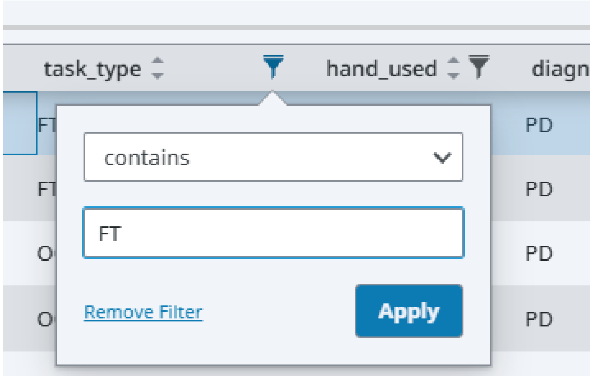
Current Chart Tags: Peak ✕

Annotations

Current Sample Tag: Select sample tag...

Current Record Tag: Select record tag...

Figure B.6: The time-series annotation application.



contains

FT

FT		PD
FT		PD
O		PD
O		PD

Figure B.7: Filtering functionality for table columns

B.6.2 Deployment

The deployment of the annotation tool utilizes Docker and Docker Compose, leveraging containerisation for a streamlined setup process. With just a single command (docker-compose up) and the docker-compose file (Figure B.8), the entire application stack, including the database and the Ignition server, can be launched without needing to manually install each component locally. The application and configuration of the Gateway can be restored from a backup file. Currently the main technical hurdle is that the database will need to be populated by the user. Otherwise, this approach ensures that the application runs identically across different systems.

```yaml
services:
  # Define the Ignition Gateway service
  gateway:
    image: inductiveautomation/ignition:latest # Use the latest version
    ports:
      - 9088:8088 # Map external port 9088 to internal port 8088
      - 9043:8043 # Map external port 9043 to internal port 8043
    networks:
      backend:
        aliases:
          - ignition # Alias for the service within the network
    volumes:
      - gw-data:/usr/local/bin/ignition/data # Mount volume for gateway data
    environment:
      - ACCEPT_IGNITION_EULA=Y
      - GATEWAY_ADMIN_USERNAME=admin
      - GATEWAY_ADMIN_PASSWORD=password
      - IGNITION_EDITION=maker
    command: >
      -n ignition_gateway # Command to run the Ignition Gateway

  # Define the database service
  db:
    image: mariadb:10.10.2 # Use the specified MariaDB version
    ports:
      - 3306:3306 # Map external port 3306 to internal port 3306
    volumes:
      - db-data:/var/lib/mysql # Mount volume for database data
    networks:
      backend:
        aliases:
          - main-db # Alias for the service within the network
    environment:
      - MARIADB_ROOT_PASSWORD=password
      - MARIADB_DATABASE=ruijin
      - MARIADB_USER=admin
      - MARIADB_PASSWORD=password

# Define the network
networks:
  backend:

# Define the volumes
volumes:
  gw-data:
  db-data:
```

Figure B.8: Docker Compose configuration for Ignition Gateway and MariaDB

B.7 Conclusion

In this appendix chapter, an overview of a custom application developed for the manual labelling of time-series data has been presented. The development of this application was motivated by the limitations identified in existing open-source solutions, which were found to be inadequate for the specific requirements of this study.

Several key contributions can be highlighted from this work:

1. A flexible database schema design has been proposed, which can be adapted to accommodate various types of time-series data and annotation schemes, allowing for adaptability to different research contexts.
2. The utilization of a low-code platform (Ignition) for rapid application development has been demonstrated. This approach illustrates how such tools can be leveraged in research settings to create bespoke solutions efficiently.
3. An architecture supporting collaborative development and annotation has been implemented, which may be particularly beneficial for interdisciplinary research teams.

While Ignition was used for this specific implementation, it should be noted that the principles and architecture discussed here could be applied using other low-code platforms or development frameworks. The approach taken emphasises the importance of creating tools that are not only functional but also adaptable and conducive to team collaboration.

The potential benefits of investing in custom tool development in research contexts have been highlighted, particularly when existing solutions are found to be inadequate. By sharing this approach, it is hoped that other researchers will be inspired to consider similar strategies when faced with unique data handling and annotation challenges in their own studies.

Future work could involve investigating the data acquisition capabilities of Ignition, particularly its ability to handle higher frequency sampling rates. While tag changes are typically expected to occur every second in the current implementation, the software's performance with more frequent data updates remains to be evaluated.

Appendix C

Reach and Grasp Investigation

This appendix provides a summary of the research conducted on the assessment of cognitive impairment through an upper limb prehension task. Due to data quality concerns and COVID-19 pandemic, the work was not fully completed; however, a summary of the findings is presented in this appendix.

The research was conducted in two stages:

1. Initially, further investigation was carried out on the dataset collected by Cosgrove et al. [40], with a primary focus on the development of grasp formation during upper limb prehension, as measured by a flexion glove sensor.
2. Subsequently, a new study was conducted in collaboration with Ruijin Hospital, examining potential abnormalities in visual attention during reach and grasp tasks.

C.1 Background

The literature on reaching and grasping movements in PD has consistently revealed impairments in the coordination of reach (transport), and grasp (manipulation), as well as an increased reliance on visual feedback. Seminal work by Castiello et al. [29] revealed that /glspd patients exhibit a delayed initiation of the manipulation component relative to the transport component, resulting in prolonged movement times. Subsequent studies from Castiello et al. [27], expanded upon this finding, demonstrating that PD patients struggle to simultaneously adapt both reach and grasp components when faced with unexpected perturbations in target position or size. The role of the basal ganglia in coordinating these complex movements was highlighted by several researchers, including Teulings et al. [184], Alberts et al. [3], and Gentilucci et al. [58]. Their work suggested that the basal ganglia dysfunction in PD leads to difficulties in performing coordinated actions, resulting in disruptions in both temporal and spatial domains of movement.

Building upon these findings, the role of visual feedback in reach and grasp movements for PwPD have also beein investigated. In [28], Castiello et al. demonstrated that removing visual feedback severly impairs the coupling between reach and grasp components in PD patients. This finding was consistent with earlier work by Jeannerod et al. [82] and supported by Jakobson et al. [79], who demonstrated that the absence of visual feedback leads to increased hand opening and earlier peak aperture, likely as a compensatory mechanism. Schettino et al. [167] further investigated this phenomenon, examining reach and grasp performance under various visual conditions. Their results reinforced the understanding that PD patients rely heavily on visual feedback to guide their movements, with performance deteriorating significantly when visual information is limited or removed.

The study conducted by Cosgrove et al. [40], investigated the interaction between cognitive decline and motor performance in PD. Their work examined reaching movements across different visual feedback conditions in PD patients with varying levels of cognitive impairment: normal cognition (PD-NC), mild cognitive impairment (PD-MCI), and dementia (PD-D). The study revealed that PD patients with dementia exhibited significantly slower reaction times across all visual feedback conditions compared to other groups, indicating a more pronounced deficit in movement planning. Furthermore, when visual feedback was removed, all PD groups showed slower movement times compared to their performance with full visual feedback. Crucially, this slowing was most pronounced in the PD-D group, suggesting that substantial cognitive decline in PD exacerbates the dependence on visual feedback during upper limb reaches. These findings demonstrate that cognitive decline not only affects traditional cognitive tasks but also significantly impacts motor performance, particularly when visual guidance is limited.

The protocol used by Cosgrove et al. [40] also incorporated the use of a flexion glove, the 5DT data glove, which offers high-resolution measurements of individual finger flexion and extension. This device potentially reveals differences in grasp formation and execution among PD subgroups that have yet to be fully explored. A review of previous work investigating this dataset revealed that the glove data had been incorrectly interpreted; the values were being read as signed integers when they were actually unsigned integers. This presented an opportunity for further investigation, particularly in exploring whether machine learning techniques could be applied to generate models capable of differentiating between the various cognitive subgroups.

C.2 Data Collection

C.2.1 Subjects

The dataset for this study was derived from the research conducted by Cosgrove, titled 'A novel diagnostic device for the objective diagnosis of Parkinson's disease with and without dementia', received National Regional Ethics Service approval (reference code 10/H1308/5) and local Research and Development approval from Leeds Teaching Hospitals NHS Trust (LTHT) (reference code UI10/9232). Subjects were measured in Leeds, UK.

The dataset includes 58 PD patients recruited from neurology clinics at Leeds Teaching Hospitals NHS Trust, and 29 healthy control subjects, primarily spouses and friends of /glspd patients. Data collection occurred between February and October 2014.

Patients were categorised into three groups based on cognitive status using the MoCA and Clinical Dementia Rating Scale (CDR) scales. Three patients with borderline scores were excluded from specific cognitive categories, resulting in the following groups: PD with normal cognition (PD-NC, n=22), PD with mild cognitive impairment (PD-MCI, n=23), and PD with dementia (PDD, n=10).

The MDS-UPDRS Part 3 was used to assess motor symptoms in all participants. Table C.1 provides a summary of the demographic details for each group, including age, gender distribution, handedness, disease duration, and the number of subjects in each category.

	Controls	PD-NC	PD-MCI	PDD
Age, years	63.8 (7.9, 50-75)	66.5 (9.4, 44-84)	70.0 (8.0, 47-85)	72.6 (5.3, 64-83)
Gender, M:F	4:15	16:6	14:9	6:4
Handedness, R:L	15:4	20:2	20:3	8:2
Duration disease, years	-	5.1 (3.7, 0.5-15)	5.7 (4.0, 0.5-15)	10.5 (6.4, 1.0-20)
Number subjects	29	22	23	10

Table C.1: Summary of the cognitive subgroups with demographic details (Standard deviation, range).

C.2.2 Protocol

The experimental protocol involved a series of reach and grasp actions using a cylindrical object. Participants were seated at a table with their hands in a semi-pronated position, little fingers aligned with specific table markings. A cylindrical object with an 8 cm diameter, resembling a beaker, was positioned 30 cm anterior to the subject. The task comprised reaching for the cylinder, grasping it, lifting it, and returning it to its original position. The experimental setup is illustrated in Figure C.1

The experiment was conducted under four distinct conditions, each repeated five times with both the dominant and non-dominant hand, resulting in a total of 40 repetitions per subject. These conditions were as follows:

1. **Self-guided reach at natural speed (NAT)**: Subjects were instructed to reach and grasp the object as they would naturally do at home, initiating movement upon hearing an auditory cue.

2. **Visually cued reach (VIS)**: Performed in a darkened room, participants responded to the cylinder being illuminated by a red light, accompanied by a simultaneous sound cue.

3. **Self-guided reach at maximum speed (MAX)**: Subjects were directed to reach and grasp the object as quickly as possible following the sound cue.

4. **Memory-guided reach (MEM)**: Participants closed their eyes before the task began and maintained this throughout the reaching, grasping, and replacing of the cylinder, only opening their eyes once the object was back on the table. As with the other conditions, the initiation cue was an auditory signal.

C.2.3 Equipment

C.2.3.1 Polhemus Patriot M

The Polhemus Patriot M EM tracking system [140], was used to record wrist position. Two sensors were attached to each wrist, as illustrated in Figure C.3. The magnetic transmitter, was positioned five centimetres behind the target object. The magnetic source was orientated such that the x-axis was facing the saggital plane of the subject, as depicted in Figure C.2

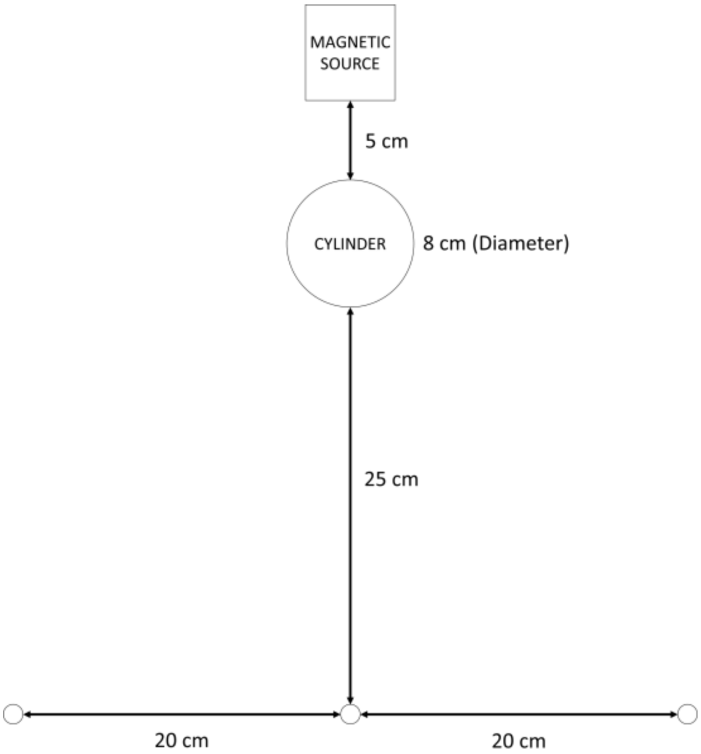


Figure C.1: Experimental setup

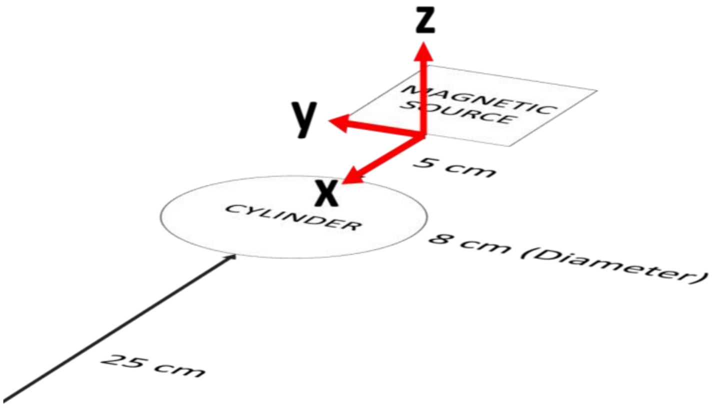


Figure C.2: Axis orientation from the perspective of the magnetic source (z axis has been inverted).

C.2.3.2 5DT Data Glove

The 5DT Data Glove Ultra 5 (Figure C.4) measures average flexion between the knuckle and first joint for each digit. This glove uses proprietary optical flex sensors, which in principle consist of a flexible tube containing optical fibres, with a light source and a photosensitive detector at opposing ends of the tube. The relative deflection of the sensor is derived from the combination of detected, direct and reflected rays [54].

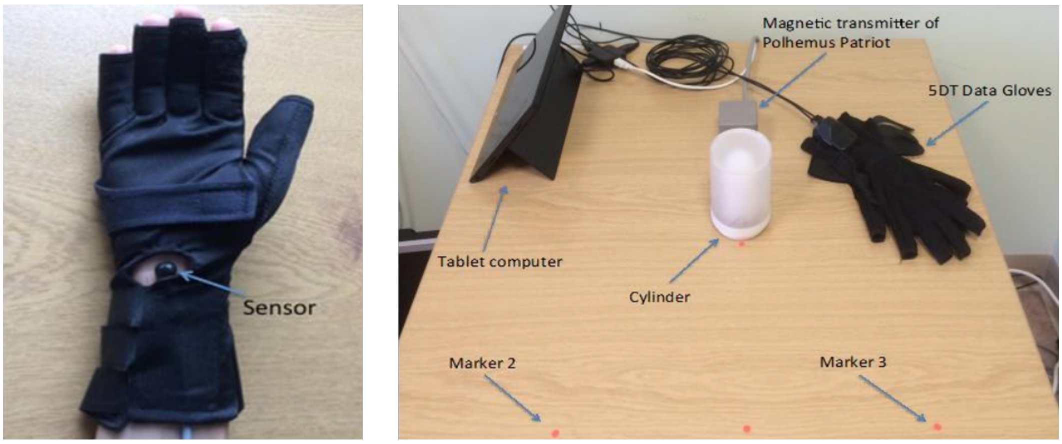


Figure C.3: Left: Polhemus Patriot M sensor placed and attached under the glove's Velcro strap. Right: The experimental configuration, depicted in C.1

Figure C.4: 5DT Data Glove Ultra 5

The glove samples data at a rate of 60 Hz, returning a 12-bit unsigned integer for each digit. A calibration process is required to map this arbitrary value to real word flexion. In this process, the range of motion for all digits is acquired by opening and closing the hand (flat to fist) several times, alternating the placement of the thumb inside and outside of the closed hand. A larger value denotes an increase in flexion. The raw value is scaled using the lower bound and dynamic range. The dynamic range is defined as the difference between the upper and lower bounds of the collected values, as depicted in Figure C.5.

The raw sensor values are then scaled using the following equation:

$$V_s = \frac{V_r - L}{D}$$

Where V_s represents the scaled (calibrated) flexion measurement, V_r is the raw sensor reading, L is the lower bound determined during calibration, and D is the dynamic range.

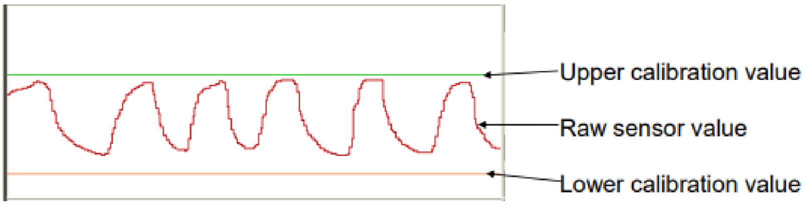


Figure C.5: Raw sensor graph, with respect to the upper and lower calibration bounds.

It is important to note that this calibration method relies on the subject's ability to achieve a full range of motion in all digits. Additionally, over time a glove may lose its factory hardware calibration, reducing the dynamic range significantly. This may be the result of movement of the opto-electronic transmitters and receivers, with respect to the fibres. In previous versions of the glove this could be tuned using several accessible potentiometers [54].

C.3 Polhemus Patriot M Analysis

Based on the experimental setup, the following positional data were anticipated: The starting x position was expected to be at minimum 38 cm, while the starting y position was predicted to be approximately 20 cm for the left hand and -20 cm for the right hand. The initial z position was anticipated to be near 0 cm. Given the setup, the minimum travel distance in the x-y plane was calculated to be 25.39 cm, and the minimum height of the lifted cylinder was expected to be 3 inches (7.62 cm).

Upon analysis of the data, it was observed that there existed two distinct distributions regarding the starting height of the hand, as depicted in Figure C.6. The experimental protocol stipulated that the initial hand height should be approximately 0 cm. However, it was surprising to discover that a significant number of recordings showed initial hand heights around -35 cm.

Figures C.7 and C.8 illustrate reach trajectories from a sample of 20 patients in each population. The black marker indicates the starting position for each attempt. Notably, the abnormal reaches exhibited an unexpected trajectory profile that did not correspond to the intended experimental setup. Instead, this profile appeared to indicate that the reach originated from the participant's lap, and that the source and cylinder were positioned closer to the edge of the table than specified in the protocol.

Following the identification of divergent trajectory profiles, further analysis was conducted to verify the consistency of these observations. It was confirmed that each participant's record-

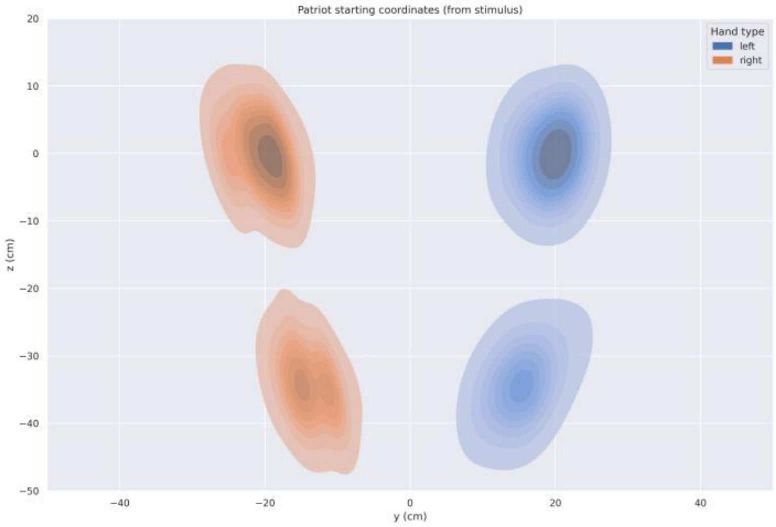


Figure C.6: Kernel density estimate (KDE) plot showing the distribution of starting points in the z-y plane.

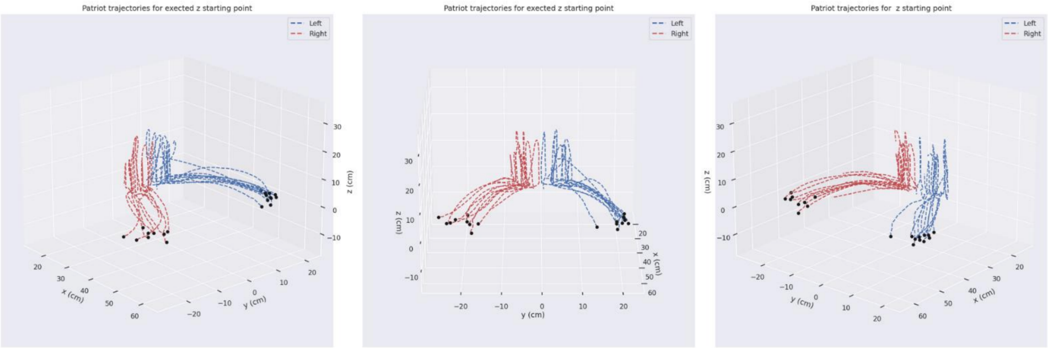


Figure C.7: Reach trajectories from participants with normal initial z-axis positions.

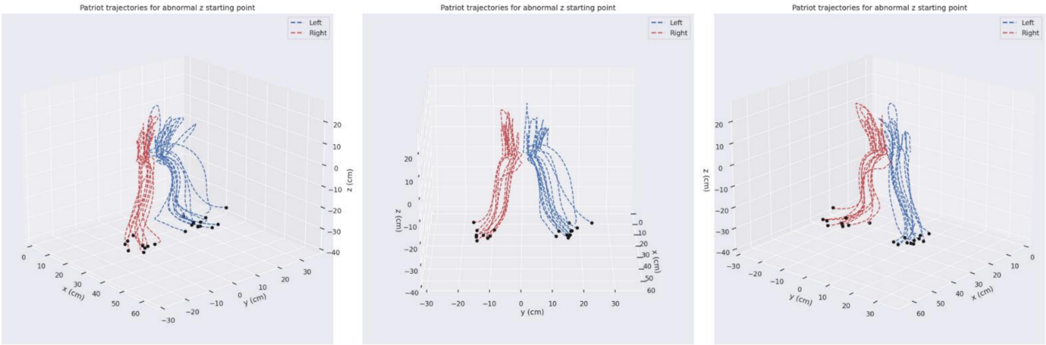


Figure C.8: Reach trajectories from participants with abnormal initial z-axis positions.

ings were consistently categorised as either normal or abnormal. Moreover, it was observed that all recordings from a given date exhibited the same classification. This anomaly affected 1099 recordings, representing 38% of subjects in the dataset who demonstrated abnormal reach trajectories that deviated from the established protocol.

C.4 5DT Data Glove analysis

The flexion sensors in both the left and right gloves exhibited significant deterioration over the duration of the study, compromising data consistency and reliability. This deterioration is clearly illustrated in Figure C.9, which depicts the maximum, minimum, and dynamic range of the index finger and thumb sensors throughout the study period. The index finger and thumb sensors were identified as the most critical for this analysis, given their role in indicating grasp formation. The observed dynamic ranges were found to be substantially lower than the 4096 possible values for the sensor, indicating a potential loss of measurement granularity. Moreover, a clear trend of declining sensor performance was observed for both gloves over the course of the study, as evidenced by the decreasing dynamic ranges. These findings raise concerns regarding the reliability of the data collected and its suitability for further analysis.

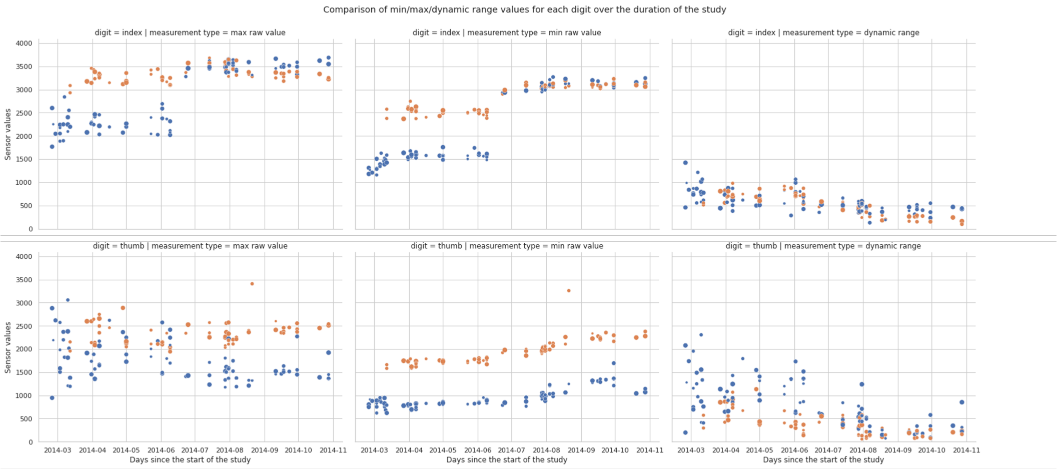


Figure C.9: The minimum, maximum and dynamic range of the index finger and thumb over the duration of the study.

C.5 Ruijin Study

In light of the data quality issues encountered in the previous study, particularly the deterioration of flexion sensors and inconsistencies in reach trajectories, a proposal was made to repeat the study following the same protocol. This repeated study was to be conducted in collaboration with Doctor Shengdi Chen and Doctor Jiang Jingwen at Ruijin Hospital, Shanghai JiaoTong University School of Medicine.

It was determined that the Polhemus Patriot M and 5DT data glove should be replaced with an optical-based hand-tracking solution, the Leap Motion Controller (LMC). This change was implemented to address the reliability concerns observed with the previous equipment.

Additionally, a new eye-tracking modality, the Pupil Core, was incorporated into the study design. The introduction of this eye-tracking technology was intended to directly quantify visual feedback, providing a more comprehensive assessment of the participants' motor and cognitive processes during the reach and grasp tasks.

C.5.1 Leap Motion Controller

The LMC was selected as a replacement for the previously used hand-tracking equipment due to its optical-based technology and potential for improved reliability. The LMC is a compact device measuring 80mm x 30mm x 13mm, equipped with two near-infrared CCD cameras and three infrared LEDs. The cameras, spaced 40 millimetres apart, operate at a resolution of 640 x 240 pixels with a refresh rate of 120 Hz. Wide-angle lenses create an interaction zone extending from 10cm to 60cm, with a 140°x120° field of view. Proprietary algorithms are employed to generate mappings from the raw sensor data to 27 key points for each hand.

Since its initial release in 2013, the LMC has undergone significant software updates, improving tracking capabilities and introducing additional features. While early versions were optimised for desktop use, subsequent iterations have been developed for head-mounted devices, specifically virtual reality headsets. The latter configuration was deemed more suitable for upper limb reach and grasp studies, as it provides a top-down view of the hand.

A study by Vysocky et. al [191] was identified as particularly relevant to the current application, as it evaluated the LMC's capabilities in a similar context. However, their research highlighted potential limitations, noting that "The LMC was not able to reliably separate the hand from the background when the hand was too close to the table surface."

To address these potential limitations, additional investigations were conducted as part of the current study. These experiments focused on the LMC's performance with different surface materials and distances. The results indicated that rubber surfaces provided better contrast for hand tracking, particularly at close range. These findings informed the experimental setup, leading to the development of an optimised protocol for LMC use in the reach and grasp tasks. Figure C.10 presents an extract from this protocol.

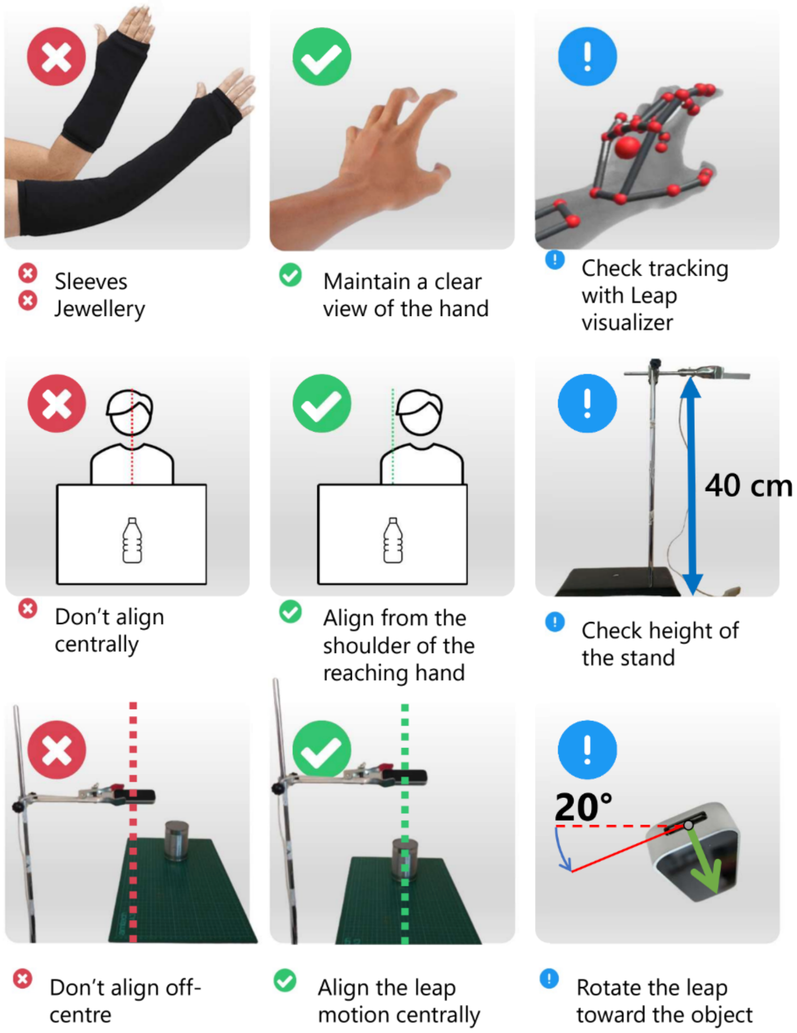

Sleeves
Jewellery
Maintain a clear view of the hand
Check tracking with Leap visualizer
Don't align centrally
Align from the shoulder of the reaching hand
40 cm
Check height of the stand
Don't align off-centre
Align the leap motion centrally
20°
Rotate the leap toward the object

Figure C.10: A extract of the protocol given to clinians in order for the leap motion to function correctly.

C.6 Pupil-Core

The incorporation of eye-tracking technology in this study was motivated by the emphasis placed on visual feedback in previous studies, and by the potential for direct quantification of visual attention to provide valuable insights into cognitive and motor processes in PD.

Previous research has demonstrated the utility of wearable eye-tracking sensors in analysing visual behavior during complex tasks. Lavoie et al. [96] observed that participants predominantly fixate on task-relevant objects, with minimal time spent fixating on their own hands during reaching tasks. Specifically, they found that participants' eyes typically arrive at objects 0.5-0.9 seconds before their hands, indicating anticipatory visual behavior that likely supports motor planning. This temporal coordination between eye and hand movements may be altered in PD due to impairments in motor planning and execution.

Furthermore, Lavoie et al.'s finding that participants briefly fixate on objects when first grasping them, but then rarely fixate on what they are holding during transport. In PD, where proprioceptive deficits are common, this pattern might be disrupted, potentially leading to increased visual monitoring of the hand and object during movement.

Eyetracking has been previously explored in terms of gait in PD. Hunt et al. found that individuals with PD [75] exhibit more task-irrelevant fixations compared to controls during walking, suggesting less efficient visual exploration of complex environments. This finding aligns with the concept of the "Attentional Landscapes Theory" discussed by Lavoie et al., which posits that attention is automatically distributed to upcoming action locations. The increased task-irrelevant fixations in PD could indicate a disruption in this attentional landscape, which may be exacerbated in cases of PD-MCI or PD-D.

The Pupil Core wearable eye-tracking system was selected for the oculomotor assessments in this study. The system comprises several key components: infrared cameras for tracking each eye, a monocular egocentric camera (world camera), a microphone, and an Inertial Measurement Unit (IMU).

The infrared eye-tracking cameras operate at a high frame rate, capturing detailed movements of each eye. The world camera, with its wide field of view, records the subject's visual scene at a resolution of 1088x1080px. The IMU, which includes an accelerometer and a gyroscope, provides data on the subject's head movements and orientation.

The system employs sophisticated algorithms to map the subject's current gaze onto the world camera image at a rate of 120Hz. This process involves integrating data from the eye-tracking cameras and the IMU to accurately project the gaze point onto the recorded visual scene.

Two primary types of eye movements are captured by the Pupil Core system: fixations and saccades. Fixations occur when the eyes are directed at a specific point in the environment for a certain duration, while saccades are rapid movements where the eyes jump from one fixation to the next. The system's high temporal resolution allows for precise measurement of these movements.

For the analysis of visual attention, it is proposed that segmentation algorithms, such as those described by Cheng et al. [32], be utilised to mask out the object and the hand in the world camera footage. This mask would be propagated throughout the video, generating a time series of where the subject's attention was directed throughout the task. By correlating this attention data with the subject's movements and task performance, relevant features can be extracted regarding the visual strategies employed during object interaction. This process is illustrated in Figure C.11.

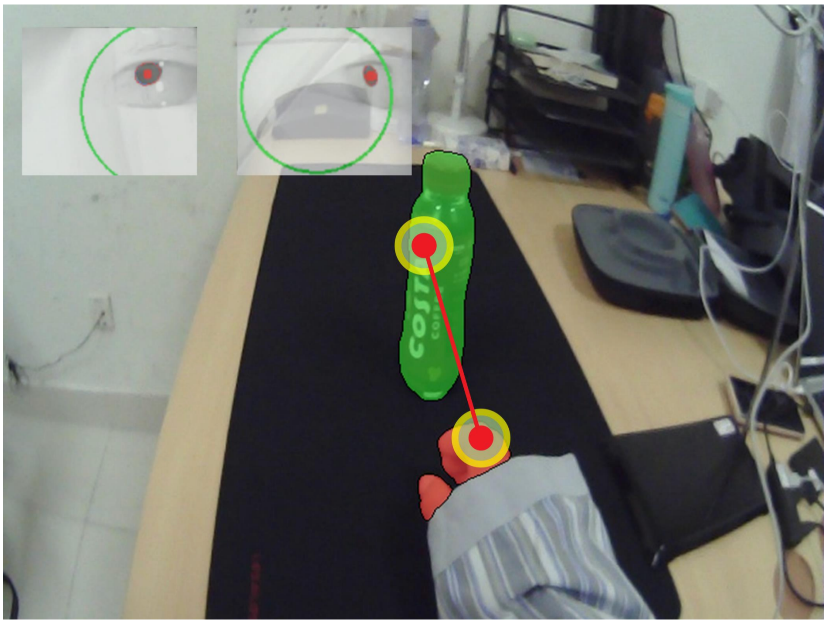


Figure C.11: An illustration of the segmentation used to identify areas of interest in the Pupil Core world camera. Note, the importance of hand visibility was subsequently reiterated using Figure C.10.

C.6.1 Outcome

Unfortunately, the study was suspended due to Covid-19 lockdowns. Prior to the suspension, data had been collected from 22 subjects with MCI. This sample size and cognitive profiles were deemed insufficient to continue with the analysis.

Future research may build upon the approach to analysing visual attention patterns outlined in this study. Such investigations could potentially provide a more nuanced understanding of how individuals with PD visually engage with their environment during upper limb prehension tasks.

List of Abbreviations

ACE-III	Addenbrooke's Cognitive Examination III.
AI	Artificial Intelligence.
ANN	Artificial Neural Network.
AUC	Area Under the Curve.
CART	Classification and Regression Trees.
CBD	Corticobasal Degeneration.
CCD	Critical Difference Diagram.
CDR	Clinical Dementia Rating Scale.
CGP	Cartesian Genetic Programming.
CNN	Convolutional Neural Network.
CSV	Comma-Separated Value.
DBS	Deep Brain Stimulation.
EM	Electromagnetic.
EMR	Electromagnetic Resonance.
FCN	Fully Convolutional Network.
FFT	Fast Fourier Transform.
FN	False Negatives.
FP	False Positives.
GLMs	Generalised Linear Models.
GUI	Graphical User Interface.
HC	Healthy Control.

ID3	Iterative Dichotomiser 3.
IMU	Inertial Measurement Unit.
L-DOPA	Levodopa.
LBD	Lewy Body Dementia.
LID	Levodopa-induced Dyskinesia.
LMC	Leap Motion Controller.
MBRS	Modified Bradykinesia Rating Scale.
MCC	Matthew's Correlation Coefficient.
MCI	Mild Cognitive Impairment.
MDS	Movement Disorder Society.
MDS-UPDRS	Movement Disorder Society - Sponsored Revision of the Unified Parkinson's Disease Rating Scale.
ML	Machine Learning.
MLE	Maximum Likelihood Estimation.
MLP	Multilayer Perceptron.
MoCA	Montreal Cognitive Assessment.
MSA	Multiple System Atrophy.
MSE	Mean Squared Error.
NC	Normal Control.
NICE	National Institute for Health and Care Excellence.
OLS	Ordinary Least Squares.
OVO	One-vs-One.
OVR	One-vs-Rest.
PD	Parkinson's Disease.
PD-NC	Parkinson's Disease Normal Cognition.
PD-D	Parkinson's Disease Dementia.
PD-MCI	Parkinson's Disease Mild Cogntive Impairment.
PSP	Progressive Supranuclar Palsy.
PwPD	People with Parkinson's Disease.
RDP	Ramer-Douglas-Peucker.
ResNet	Residual Network.

RGB	Red Green Blue.
ROC	Receiver Operating Characteristic.
ROC-AUC	Receiver Operating Characteristic Area Under the Curve.
ROCF	Rey-Osterrieth Complex Figure Test.
Rocket	RandOm Convolutional KErnel Transform.
SCADA	Supervisory Control and Data Acquisition.
SNR	Signal to Noise Ratio.
TN	True Negatives.
TP	True Positives.
TSC	Time Series Classification.
UPDRS	Unified Parkinson’s Disease Rating Scale.

References

[1] Dag Aarsland, Julia Zaccai, and Carol Brayne. A systematic review of prevalence studies of dementia in parkinson's disease. *Mov. Disord.*, 20(10):1255–1263, October 2005.

[2] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, pages 2623–2631, New York, NY, USA, July 2019. Association for Computing Machinery.

[3] J L Alberts, J R Tresilian, and G E Stelmach. The co-ordination and phasing of a bilateral prehension task. the influence of parkinson's disease. *Brain*, 121 (Pt 4):725–742, April 1998.

[4] Hugh Alderwick and Jennifer Dixon. The NHS long term plan. *BMJ*, 364:l84, January 2019.

[5] Mohamad Alissa, Michael A Lones, Jeremy Cosgrove, Jane E Alty, Stuart Jamieson, Stephen L Smith, and Marta Vallejo. Parkinson's disease diagnosis using convolutional neural networks and figure-copying tasks. *Neural Comput. Appl.*, 34(2):1433–1453, January 2022.

[6] Khaled Alomar, Halil Ibrahim Aysel, and Xiaohao Cai. Data augmentation in classification and segmentation: A survey and new strategies. *J. Imaging*, 9(2):46, February 2023.

[7] Angelo Antonini, Heinz Reichmann, Giovanni Gentile, Michela Garon, Chiara Tedesco, Anika Frank, Bjoern Falkenburger, Spyridon Konitsiotis, Konstantinos Tsamis, Georgios Rigas, Nicholas Kostikis, Adamantios Ntanis, and Constantinos Pattichis. Toward objective monitoring of parkinson's disease motor symptoms using a wearable device: wearability and performance evaluation of PDMonitor®. *Front. Neurol.*, 14:1080752, May 2023.

[8] P Armitage. Tests for linear trends in proportions and frequencies. *Biometrics*, 11(3):375–386, 1955.

[9] Baidu. Curve: An integrated experimental platform for time series data anomaly detection.

[10] Roger A Barker, Anders Björklund, Don M Gash, Alan Whone, Amber Van Laar, Jeffrey H Kordower, Krystof Bankiewicz, Karl Kieburtz, Mart Saarma, Sigrid Booms, Henri J Huttunen, Adrian P Kells, Massimo S Fiandaca, A Jon Stoessl, David Eidelberg, Howard Federoff, Merja H Voutilainen, David T Dexter, Jamie Eberling, Patrik Brundin, Lyndsey Isaacs, Leah Mursaleen, Eros Bresolin, Camille Carroll, Alasdair Coles, Brian Fiske, Helen Matthews, Codrin Lungu, Richard K Wyse, Simon Stott, and Anthony E Lang. GDNF and parkinson's disease: Where next? a summary from a recent workshop. *J. Parkinsons. Dis.*, 10(3):875–891, 2020.

[11] Lucy C Beishon, Angus P Batterham, Terry J Quinn, Christopher P Nelson, Ronney B Panerai, Thompson Robinson, and Victoria J Haunton. Addenbrooke's cognitive examination III (ACE-III) and mini-ACE for the detection of dementia and mild cognitive impairment. *Cochrane Database Syst. Rev.*, 12:CD013282, December 2019.

[12] W Birkmayer and O Hornykiewicz. [the L-3,4-dioxyphenylalanine (DOPA)-effect in parkinson-akinesia]. *Wien. Klin. Wochenschr.*, 73:787–788, November 1961.

[13] Soo Borson, James M Scanlan, Peijun Chen, and Mary Ganguli. The mini-cog as a screen for dementia: validation in a population-based sample. *J. Am. Geriatr. Soc.*, 51(10):1451–1454, October 2003.

[14] Heiko Braak, Kelly Del Tredici, Udo Rüb, Rob A I de Vos, Ernst N H Jansen Steur, and Eva Braak. Staging of brain pathology related to sporadic parkinson's disease. *Neurobiol. Aging*, 24(2):197–211, March 2003.

[15] Malte Brammerloh, Evgeniya Kirilina, Anneke Alkemade, Pierre-Louis Bazin, Caroline Jantzen, Carsten Jäger, Andreas Herrler, Kerrin J Pine, Penny A Gowland, Markus Morawski, Birte U Forstmann, and Nikolaus Weiskopf. Swallow tail sign: Revisited. *Radiology*, 305(3):674–677, December 2022.

[16] Leo Breiman. Random forests. *Mach. Learn.*, 45(1):5–32, October 2001.

[17] Leo Breiman, Jerome H Friedman, Richard A Olshen, and Charles J Stone. *Classification And Regression Trees.* Routledge, 1st edition edition, October 2017.

[18] Henry Brodaty, Dimity Pond, Nicola M Kemp, Georgina Luscombe, Louise Harding, Karen Berman, and Felicia A Huppert. The GPCOG: a new screening test for dementia designed for general practice. *J. Am. Geriatr. Soc.*, 50(3):530–534, March 2002.

[19] A H Butt, E Rovini, C Dolciotti, P Bongioanni, G De Petris, and F Cavallo. Leap motion evaluation for assessment of upper limb motor skills in parkinson's disease. In *2017 International Conference on Rehabilitation Robotics (ICORR)*, pages 116–121, July 2017.

[20] A H Butt, E Rovini, C Dolciotti, G De Petris, P Bongioanni, M C Carboncini, and F Cavallo. Objective and automatic classification of parkinson disease with leap motion controller. *Biomed. Eng. Online*, 17(1):168, November 2018.

[21] Abdul Haleem Butt, Erika Rovini, Hamido Fujita, Carlo Maremmani, and Filippo Cavallo. Data-driven models for objective grading improvement of parkinson's disease. *Ann. Biomed. Eng.*, 48(12):2976–2987, December 2020.

[22] S Butterworth. On the theory of filter amplifiers. *Experimental Wireless & the Wireless Engineer*, 7:536–541, October 1930.

[23] Davide Maria Cammisuli, Roberto Ceravolo, and Ubaldo Bonuccelli. Non-pharmacological interventions for parkinson's disease mild cognitive impairment: future directions for research. *Neural Regeneration Res.*, 15(9):1650–1651, September 2020.

[24] R O Canham, S L Smith, and A M Tyrrell. Automated scoring of a neuropsychological test: the rey osterrieth complex figure. In *Proceedings of the 26th Euromicro Conference. EUROMICRO 2000. Informatics: Inventing the Future*, volume 2, pages 406–413 vol.2. IEEE, 2000.

[25] A Carlsson. Treatment of parkinson's with L-DOPA. the early discovery phase, and a comment on current problems. *J. Neural Transm.*, 109(5-6):777–787, May 2002.

[26] Camille B Carroll, Douglas Webb, Kara Nicola Stevens, Jane Vickery, Vicky Eyre, Susan Ball, Richard Wyse, Mike Webber, Andy Foggo, John Zajicek, Alan Whone, and Siobhan Creanor. Simvastatin as a neuroprotective treatment for parkinson's disease (PD STAT): protocol for a double-blind, randomised, placebo-controlled futility study. *BMJ Open*, 9(10):e029740, October 2019.

[27] U Castiello, K Bennett, C Bonfiglioli, S Lim, and R F Peppard. The reach-to-grasp movement in parkinson's disease: response to a simultaneous perturbation of object position and object size. *Exp. Brain Res.*, 125(4):453–462, April 1999.

[28] U Castiello, K M Bennett, and C Mucignat. The reach to grasp movement of blind subjects. *Exp. Brain Res.*, 96(1):152–162, 1993.

[29] U Castiello, G E Stelmach, and A N Lieberman. Temporal dissociation of the prehension pattern in parkinson's disease. *Neuropsychologia*, 31(4):395–402, April 1993.

[30] Jean Martin Charcot. *Lecons sur, les maladies du système nerveux.* Lecrosnier et Babé, 1886.

[31] Kai-Hsiang Chen, Po-Chieh Lin, Bing-Shiang Yang, and Yu-Jung Chen. The difference in visuomotor feedback velocity control during spiral drawing between parkinson's disease and essential tremor. *Neurol. Sci.*, 39(6):1057–1063, June 2018.

[32] Ho Kei Cheng, Yu-Wing Tai, and Chi-Keung Tang. Modular interactive video object segmentation: Interaction-to-mask, propagation and difference-aware fusion. In *CVPR*, 2021.

[33] Davide Chicco and Giuseppe Jurman. The advantages of the matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics*, 21(1):6, January 2020.

[34] Davide Chicco, Niklas Tötsch, and Giuseppe Jurman. The matthews correlation coefficient (MCC) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation. *BioData Min.*, 14(1):13, February 2021.

[35] Carl E Clarke, Smitaa Patel, Natalie Ives, Caroline E Rick, Rebecca Woolley, Keith Wheatley, Marion F Walker, Shihua Zhu, Rebecca Kandiyali, Guiqing Yao, Catherine M Sackley, and on behalf of the PD REHAB Collaborative Group. *UK Parkinson's Disease Society Brain Bank Diagnostic Criteria.* NIHR Journals Library, August 2016.

[36] Robert J Coffey. Deep brain stimulation devices: a brief technical history and review. *Artif. Organs*, 33(3):208–220, March 2009.

[37] Yaroslau Compta, Laura Parkkinen, Sean S O'Sullivan, Jana Vandrovcova, Janice L Holton, Catherine Collins, Tammaryn Lashley, Constantinos Kallis, David R Williams, Rohan de Silva, Andrew J Lees, and Tamas Revesz. Lewy- and alzheimer-type pathologies in parkinson's disease dementia: which is more important? *Brain*, 134(Pt 5):1493–1505, May 2011.

[38] James W Cooley and John W Tukey. An algorithm for the machine calculation of complex fourier series. *Math. Comput.*, 19(90):297–301, 1965.

[39] Jeremy Cosgrove. Investigating reach and grasp in parkinson's disease cognitive impairment.

[40] Jeremy Cosgrove, Mark R Hinder, Rebecca J St George, Chiara Picardi, Stephen L Smith, Michael A Lones, Stuart Jamieson, and Jane E Alty. Significant cognitive decline in parkinson's disease exacerbates the reliance on visual feedback during upper limb reaches. *Neuropsychologia*, 157:107885, July 2021.

[41] G C Cotzias, M H Van Woert, and L M Schiffer. Aromatic amino acids and modification of parkinsonism. *N. Engl. J. Med.*, 276(7):374–379, February 1967.

[42] Jonas Jardim de Paula, Mônica Vieira Costa, Giovanna de Freitas de Andrade, Rafaela Teixeira Ávila, and Leandro Fernandes Malloy-Diniz. Validity and reliability of a "simplified" version of the taylor complex figure test for the assessment of older adults with low formal education. *Dement Neuropsychol*, 10(1):52–57, 2016.

[43] Nele Demeyere, Marleen Haupt, Sam S Webb, Lea Strobel, Elise T Milosevich, Margaret J Moore, Hayley Wright, Kathrin Finke, and Mihaela D Duta. Introducing the tablet-based oxford cognitive screen-plus (OCS-plus) as an assessment tool for subtle cognitive impairments. *Sci. Rep.*, 11(1):8000, April 2021.

[44] Angus Dempster, François Petitjean, and Geoffrey I Webb. ROCKET: Exceptionally fast and accurate time series classification using random convolutional kernels. *arXiv [cs.LG]*, October 2019.

[45] Angus Dempster, Daniel F Schmidt, and Geoffrey I Webb. MINIROCKET: A very fast (almost) deterministic transform for time series classification. *arXiv [cs.LG]*, December 2020.

[46] Angus Dempster, Daniel F Schmidt, and Geoffrey I Webb. MINIROCKET: A very fast (almost) deterministic transform for time series classification. *arXiv [cs.LG]*, December 2020.

[47] Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *J. Mach. Learn. Res.*, 7:1–30, December 2006.

[48] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255. IEEE, June 2009.

[49] T G Dietterich. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Comput.*, 10(7):1895–1923, September 1998.

[50] Ruth Djaldetti, Ilan Ziv, and Eldad Melamed. The mystery of motor asymmetry in parkinson's disease. *Lancet Neurol.*, 5(9):796–802, September 2006.

[51] E Ray Dorsey, Todd Sherer, Michael S Okun, and Bastiaan R Bloem. The emerging evidence of the parkinson pandemic. *J. Parkinsons. Dis.*, 8(s1):S3–S8, 2018.

[52] Peter Drotár, Jiří Mekyska, Zdeněk Smékal, Irena Rektorová, Lucia Masarová, and Marcos Faundez-Zanuy. Contribution of different handwriting modalities to differential diagnosis of parkinson's disease. *arXiv [eess.SP]*, March 2022.

[53] Claudia Ferraris, Roberto Nerino, Antonio Chimienti, Giuseppe Pettiti, Nicola Cau, Veronica Cimolin, Corrado Azzaro, Giovanni Albani, Lorenzo Priano, and Alessandro Mauro. A self-managed system for automated assessment of UPDRS upper limb tasks in parkinson's disease. *Sensors*, 18(10):3523, October 2018.

[54] Fifth Dimension Technologies. *5DT data glove ultra series user's manual*, October 2004.

[55] Christopher Frank, Giovanna Pari, and John P Rossiter. Approach to diagnosis of parkinson disease. *Can. Fam. Physician*, 52(7):862–868, July 2006.

[56] Chao Gao, Stephen Smith, Michael Lones, Stuart Jamieson, Jane Alty, Jeremy Cosgrove, Pingchen Zhang, Jin Liu, Yimeng Chen, Juanjuan Du, Shishuang Cui, Haiyan Zhou, and Shengdi Chen. Objective assessment of bradykinesia in parkinson's disease using evolutionary algorithms: clinical validation. *Transl. Neurodegener.*, 7:18, August 2018.

[57] GBD 2016 Neurology Collaborators. Global, regional, and national burden of neurological disorders, 1990-2016: a systematic analysis for the global burden of disease study 2016. *Lancet Neurol.*, 18(5):459–480, May 2019.

[58] M Gentilucci and A Negrotti. Planning and executing an action in parkinson's disease. *Mov. Disord.*, 14(1):69–79, January 1999.

[59] Christopher G Goetz, Werner Poewe, Olivier Rascol, Cristina Sampaio, Glenn T Stebbins, Carl Counsell, Nir Giladi, Robert G Holloway, Charity G Moore, Gregor K Wenning, Melvin D Yahr, Lisa Seidl, and Movement Disorder Society Task Force on Rating Scales for Parkinson's Disease. Movement disorder society task force report on the hoehn and yahr staging scale: status and recommendations. *Mov. Disord.*, 19(9):1020–1028, September 2004.

[60] Christopher G Goetz and Glenn T Stebbins. Assuring interrater reliability for the UPDRS motor section: utility of the UPDRS teaching tape. *Mov. Disord.*, 19(12):1453–1456, December 2004.

[61] Christopher G Goetz, Barbara C Tilley, Stephanie R Shaftman, Glenn T Stebbins, Stanley Fahn, Pablo Martinez-Martin, Werner Poewe, Cristina Sampaio, Matthew B Stern, Richard Dodel, Bruno Dubois, Robert Holloway, Joseph Jankovic, Jaime Kulisevsky, Anthony E Lang, Andrew Lees, Sue Leurgans, Peter A LeWitt, David Nyenhuis, C Warren Olanow, Olivier Rascol, Anette Schrag, Jeanne A Teresi, Jacobus J van Hilten, Nancy LaPelle, and Movement Disorder Society UPDRS Revision Task Force. Movement disorder society-sponsored revision of the unified parkinson's disease rating

scale (MDS-UPDRS): scale presentation and clinimetric testing results. *Mov. Disord.*, 23(15):2129–2170, November 2008.

[62] Sylvain Gugger and Jeremy Howard. AdamW and super-convergence is now the fastest way to train neural nets. `https://www.fast.ai/posts/2018-07-02-adam-weight-decay.html`, July 2018. Accessed: 2023-8-21.

[63] Juan Haladjian. The wearables development toolkit: An integrated development environment for activity recognition applications. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 3(4):1–26, December 2019.

[64] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition (Springer Series in Statistics)*. Springer, 2 edition, February 2009.

[65] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv [cs.CV]*, December 2015.

[66] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. *arXiv [cs.CV]*, February 2015.

[67] Dustin A Heldman, Joseph P Giuffrida, Robert Chen, Megan Payne, Filomena Mazzella, Andrew P Duker, Alok Sahay, Sang Jin Kim, Fredy J Revilla, and Alberto J Espay. The modified bradykinesia rating scale for parkinson's disease: reliability and comparison with kinematic measures. *Mov. Disord.*, 26(10):1859–1863, August 2011.

[68] Neal Hermanowicz, Sarah A Jones, and Robert A Hauser. Impact of non-motor symptoms in parkinson's disease: a PMDAlliance survey. *Neuropsychiatr. Dis. Treat.*, 15:2205–2212, August 2019.

[69] M M Hoehn and M D Yahr. Parkinsonism: onset, progression and mortality. *Neurology*, 17(5):427–442, May 1967.

[70] Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 42(1):80, February 2000.

[71] Holistics Software. dbdiagram.io. `https://dbdiagram.io/home`, 2024. Accessed: 2024-7-22.

[72] Oleh Hornykiewicz. Basic research on dopamine in parkinson's disease and the discovery of the nigrostriatal dopamine pathway: the view of an eyewitness. *Neurodegener. Dis.*, 5(3-4):114–117, March 2008.

[73] Jeremy Howard and Sylvain Gugger. fastai: A layered API for deep learning. *arXiv [cs.LG]*, February 2020.

[74] A J Hughes, S E Daniel, L Kilford, and A J Lees. Accuracy of clinical diagnosis of idiopathic parkinson's disease: a clinico-pathological study of 100 cases. *J. Neurol. Neurosurg. Psychiatry*, 55(3):181–184, March 1992.

[75] David Hunt, Samuel Stuart, Jeremy Nell, Jeffrey M Hausdorff, Brook Galna, Lynn Rochester, and Lisa Alcock. Do people with parkinson's disease look at task relevant stimuli when walking? an exploration of eye movements. *Behav. Brain Res.*, 348:82–89, August 2018.

[76] John D Hunter. Matplotlib: A 2D graphics environment. *Comput. Sci. Eng.*, 9(3):90–95, May 2007.

[77] Md Saiful Islam, Wasifur Rahman, Abdelrahman Abdelkader, Sangwu Lee, Phillip T Yang, Jennifer Lynn Purks, Jamie Lynn Adams, Ruth B Schneider, Earl Ray Dorsey, and Ehsan Hoque. Using AI to measure parkinson's disease severity at home. *NPJ Digit Med*, 6(1):156, August 2023.

[78] Hassan Ismail Fawaz, Benjamin Lucas, Germain Forestier, Charlotte Pelletier, Daniel F Schmidt, Jonathan Weber, Geoffrey I Webb, Lhassane Idoumghar, Pierre-Alain Muller, and François Petitjean. InceptionTime: Finding AlexNet for time series classification. *Data Min. Knowl. Discov.*, 34(6):1936–1962, November 2020.

[79] L S Jakobson and M A Goodale. Factors affecting higher-order movement planning: a kinematic analysis of human prehension. *Exp. Brain Res.*, 86(1):199–208, 1991.

[80] Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, and Jonathan Taylor. *An Introduction to Statistical Learning: With Applications in Python*. Springer Nature, Cham, Switzerland, August 2023.

[81] J Jankovic. Parkinson's disease: clinical features and diagnosis. *J. Neurol. Neurosurg. Psychiatry*, 79(4):368–376, April 2008.

[82] M Jeannerod. The timing of natural prehension movements. *J. Mot. Behav.*, 16(3):235–254, September 1984.

[83] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, June 2014.

[84] Siddharth Kaul and Rodger J Elble. Impaired pentagon drawing is an early predictor of cognitive decline in parkinson's disease. *Mov. Disord.*, 29(3):427–428, March 2014.

[85] Angie A Kehagia, Roger A Barker, and Trevor W Robbins. Cognitive impairment in parkinson's disease: the dual syndrome hypothesis. *Neurodegener. Dis.*, 11(2):79–92, 2013.

[86] Taha Khan, Dag Nyholm, Jerker Westin, and Mark Dougherty. A computer vision framework for finger-tapping evaluation in parkinson's disease. *Artif. Intell. Med.*, 60(1):27–40, January 2014.

[87] J Kiefer and J Wolfowitz. Stochastic estimation of the maximum of a regression function. *Ann. Math. Stat.*, 23(3):462–466, September 1952.

[88] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv [cs.LG]*, December 2014.

[89] K Kosaka, S Oyanagi, M Matsushita, and A Hori. Presenile dementia with alzheimer-, pick- and lewy-body changes. *Acta Neuropathol.*, 36(3):221–233, November 1976.

[90] Kosmas Kritsis, Maximos Kaliakatsos-Papakostas, Vassilis Katsouros, and Aggelos Pikrakis. Deep convolutional and LSTM neural network architectures on leap motion hand tracking data sequences. In *2019 27th European Signal Processing Conference (EUSIPCO)*, pages 1–5, September 2019.

[91] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet classification with deep convolutional neural networks. In F Pereira, C J Burges, L Bottou, and K Q Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

[92] L Kurlowicz and M Wallace. The mini mental state examination (MMSE). *Director*, 7(2):62, 1999.

[93] Emily S Kuschner, Kimberly E Bodner, and Nancy J Minshew. Local vs. global approaches to reproducing the rey osterrieth complex figure by children, adolescents, and adults with high-functioning autism. *Autism Res.*, 2(6):348–358, December 2009.

[94] Stuart E Lacy, Stephen L Smith, and Michael A Lones. Using echo state networks for classification: A case study in parkinson's disease diagnosis. *Artif. Intell. Med.*, 86:53–59, March 2018.

[95] José L Lanciego, Natasha Luquin, and José A Obeso. Functional neuroanatomy of the basal ganglia. *Cold Spring Harb. Perspect. Med.*, 2(12):a009621, December 2012.

[96] Ewen B Lavoie, Aïda M Valevicius, Quinn A Boser, Ognjen Kovic, Albert H Vette, Patrick M Pilarski, Jacqueline S Hebert, and Craig S Chapman. Using synchronized

eye and motion tracking to determine high-precision eye-movement patterns during object-interaction tasks. *J. Vis.*, 18(6):18, June 2018.

[97] Stanley Lemeshow, Rodney X Sturdivant, and David W Hosmer, Jr. *Applied Logistic Regression.* Wiley & Sons, Limited, John, 2013.

[98] F H Lewy. Zue pathologischen anatomie der paralysis agitans. *Detsch Zeitschr fur Nervenhaeilkunde*, 50:50, 1913.

[99] Junjie Li, Huaiyu Zhu, Yun Pan, Haotian Wang, Zhidong Cen, Dehao Yang, and Wei Luo. Three-dimensional pattern features in finger tapping test for patients with parkinson's disease. *Conf. Proc. IEEE Eng. Med. Biol. Soc.*, 2020:3676–3679, July 2020.

[100] Lei Li, Hongbo Fu, and Chiew-Lan Tai. Fast sketch segmentation and labeling with deep learning. *IEEE Comput. Graph. Appl.*, 39(2):38–51, 2019.

[101] Tianbai Li and Weidong Le. Biomarkers for parkinson's disease: How good are they? *Neurosci. Bull.*, 36(2):183–194, February 2020.

[102] Irene Litvan, Dag Aarsland, Charles H Adler, Jennifer G Goldman, Jaime Kulisevsky, Brit Mollenhauer, Maria C Rodriguez-Oroz, Alexander I Tröster, and Daniel Weintraub. MDS task force on mild cognitive impairment in parkinson's disease: critical review of PD-MCI. *Mov. Disord.*, 26(10):1814–1824, August 2011.

[103] Xiaolong Liu, Zhidong Deng, and Yuhan Yang. Recent progress in semantic image segmentation. *Artificial Intelligence Review*, 52(2):1089–1106, August 2019.

[104] Michael A Lones. How to avoid machine learning pitfalls: a guide for academic researchers. *arXiv [cs.LG]*, August 2021.

[105] Lu Lu, Yeonjong Shin, Yanhui Su, and George Em Karniadakis. Dying ReLU and initialization: Theory and numerical examples. *arXiv [stat.ML]*, March 2019.

[106] Andrew L Maas. Rectifier nonlinearities improve neural network acoustic models. *W&CP*, 28, 2013.

[107] Luca Marsili, Giovanni Rizzo, and Carlo Colosimo. Diagnostic criteria for parkinson's disease: From james parkinson to the concept of prodromal disease. *Front. Neurol.*, 9:156, March 2018.

[108] Pablo Martınez-Martın, Carmen Rodríguez-Blázquez, Mónica Kurtis, and K. Ray Chaudhuri. The impact of non-motor symptoms on health-related quality of life of patients with parkinson's disease. *Movement Disorders*, 2011.

[109] Alexander Mathis, Pranav Mamidanna, Kevin M Cury, Taiga Abe, Venkatesh N Murthy, Mackenzie W Mathis, and Matthias Bethge. DeepLabCut: markerless pose estimation of user-defined body parts with deep learning. *Nature Neuroscience*, 2018.

[110] Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *Bull. Math. Biophys.*, 5(4):115–133, December 1943.

[111] John E Meyers and Kelly R Meyers. *Rey complex figure test and recognition trial supplemental norms for children and adults.* Psychological Assessment Resources, Odessa, Flor., 1996.

[112] Microsoft. TagAnomaly: Anomaly detection analysis and labeling tool.

[113] Marvin Minsky and Seymour Papert. *Perceptrons: An Introduction to Computational Geometry.* MIT Press, Cambridge, MA, USA, 1969.

[114] Gareth Morinan, Yuriy Dushin, Grzegorz Sarapata, Samuel Rupprechter, Yuwei Peng, Christine Girges, Maricel Salazar, Catherine Milabo, Krista Sibley, Thomas Foltynie, Ioana Cociasu, Lucia Ricciardi, Fahd Baig, Francesca Morgante, Louise-Ann Leyland, Rimona S Weil, Ro'ee Gilron, and Jonathan O'Keeffe. Computer vision quantification of whole-body parkinsonian bradykinesia using a large multi-site population. *NPJ Parkinsons Dis*, 9(1):10, January 2023.

[115] Elena Moro, Michael Schüpbach, Tobias Wächter, Niels Allert, Roberto Eleopra, Christopher R Honey, Mauricio Rueda, Mya C Schiess, Yasushi Shimo, Peter Valkovic, Alan Whone, and Herman Stoevelaar. Referring parkinson's disease patients for deep brain stimulation: a RAND/UCLA appropriateness study. *J. Neurol.*, 263(1):112–119, January 2016.

[116] John C Morris, Sandra Weintraub, Helena C Chui, Jeffrey Cummings, Charles Decarli, Steven Ferris, Norman L Foster, Douglas Galasko, Neill Graff-Radford, Elaine R Peskind, Duane Beekly, Erin M Ramos, and Walter A Kukull. The uniform data set (UDS): clinical and cognitive variables and descriptive data from alzheimer disease centers. *Alzheimer Dis. Assoc. Disord.*, 20(4):210–216, 2006.

[117] Anastasia Moshkova, Vladislav Bukin, and Andrey Samorodov. The error evaluation of the amplitude of finger tapping and pronation/supination of the palm exercises recorded by leap motion. In *2023 Systems and Technologies of the Digital HealthCare (STDH)*, pages 81–83. IEEE, October 2023.

[118] Anastasia Moshkova, Andrey Samorodov, Natalia Voinova, Alexander Volkov, Ekaterina Ivanova, and Ekaterina Fedotova. Parkinson's disease detection by using machine

learning algorithms and hand movement signal from LeapMotion sensor. In *2020 26th Conference of Open Innovations Association (FRUCT)*. IEEE, April 2020.

[119] Anastasia A Moshkova, Andrey V Samorodov, Natalia A Voinova, Ekaterina O Ivanova, and Ekaterina Y Fedotova. Hand movement kinematic parameters assessment for parkinson's disease patients. In *2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus)*, pages 2836–2841. IEEE, January 2021.

[120] Siti Anizah Muhamed, Rachel Newby, Stephen L Smith, and Peter Kempster. Objective evaluation of bradykinesia in parkinson's disease using evolutionary algorithms. In *11th International Conference on Bio-inspired Systems and Signal Processing*, pages 63–69. unknown, January 2018.

[121] Daniel L Murman. Early treatment of parkinson's disease: opportunities for managed care. *Am. J. Manag. Care*, 18(7 Suppl):S183–8, September 2012.

[122] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. *ICML*, pages 807–814, June 2010.

[123] Ziad S Nasreddine, Natalie A Phillips, Valérie Bédirian, Simon Charbonneau, Victor Whitehead, Isabelle Collin, Jeffrey L Cummings, and Howard Chertkow. The montreal cognitive assessment, MoCA: a brief screening tool for mild cognitive impairment. *J. Am. Geriatr. Soc.*, 53(4):695–699, April 2005.

[124] Izaak Neutelings. Convolutional operator. `https://tikz.net/conv2d/`, 2021. Accessed: 2023-8-5.

[125] Izaak Neutelings. Neural networks. `https://tikz.net/neural_networks/`, 2021. Accessed: 2023-8-5.

[126] NHS. Parkinson's disease - symptoms. `https://www.nhs.uk/conditions/parkinsons-disease/symptoms/`. Accessed: 2023-8-16.

[127] NICE. Devices for remote monitoring of parkinnsonn's disease (2023) diagnostics guidance DG51, January 2023.

[128] Arani Nitkunan, Croydon University Hospital, Joanne Lawrence, Mary M Reilly, Association of British Neurologists, and UCL Institute of Neurology. Association of british neurologists: UK neurology workforce survey. *Adv. Clin. Neurosci. Rehabil.*, 20(1):28–32, December 2020.

[129] Ignacio Oguiza. tsai - a state-of-the-art deep learning library for time series and sequential data. Github, 2022.

[130] Ayse Betul Oktay and Abdulkadir Kocer. Differential diagnosis of parkinson and essential tremor with convolutional LSTM networks. *Biomed. Signal Process. Control*, 56:101683, February 2020.

[131] P A Osterrieth. Le test de copie d'une figure complexe; contribution à l'étude de la perception et de la mémoire. *Arch. Psychol.*, 30:206–356, 1944.

[132] Sujith Ovallath and P Deepa. The history of parkinsonism: descriptions in ancient indian medical literature. *Mov. Disord.*, 28(5):566–568, May 2013.

[133] Ingyu Park, Yun Joong Kim, Yeo Jin Kim, and Unjoo Lee. Automatic, qualitative scoring of the interlocking pentagon drawing test (PDT) based on U-net and mobile sensor data. *Sensors*, 20(5), February 2020.

[134] Ingyu Park and Unjoo Lee. Automatic, qualitative scoring of the clock drawing test (CDT) based on U-net, CNN and mobile sensor data. *Sensors*, 21(15), August 2021.

[135] James Parkinson. An essay on the shaking palsy. 1817. *J. Neuropsychiatry Clin. Neurosci.*, 14(2):223–36; discussion 222, 2002.

[136] F Pedregosa, G Varoquaux, A Gramfort, V Michel, B Thirion, O Grisel, M Blondel, P Prettenhofer, R Weiss, V Dubourg, J Vanderplas, A Passos, D Cournapeau, M Brucher, M Perrot, and E Duchesnay. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.*

[137] Fabian Pedregosa, G Varoquaux, Alexandre Gramfort, V Michel, B Thirion, O Grisel, Mathieu Blondel, Gilles Louppe, P Prettenhofer, Ron Weiss, Ron J Weiss, J Vanderplas, Alexandre Passos, D Cournapeau, M Brucher, M Perrot, and E Duchesnay. Scikit-learn: Machine learning in python. *J. Mach. Learn. Res.*, abs/1201.0490, February 2011.

[138] Jeremy Petch, Shuang Di, and Walter Nelson. Opening the black box: The promise and limitations of explainable machine learning in cardiology. *Can. J. Cardiol.*, 38(2):204–213, February 2022.

[139] Marco A Petilli, Roberta Daini, Francesca Lea Saibene, and Marco Rabuffetti. Automated scoring for a tablet-based rey figure copy task differentiates constructional, organisational, and motor abilities. *Sci. Rep.*, 11(1):14895, July 2021.

[140] Polhemus. PATRIOT M USER MANUAL. `https://ftp.polhemus1.com/pub/Trackers/Patriot_M/Manuals/PATRIOT_M_User_Manual_URM11PH273-B.pdf`, 2013.

[141] Marius-Constantin Popescu, Valentina E Balas, Liliana Perescu-Popescu, and Nikos Mastorakis. Multilayer perceptron and neural networks. *WSEAS Trans. Cir. and Sys.*, 8(7):579–588, July 2009.

[142] Amir Poreh, Jennifer B Levin, and Max Teaford. Geriatric complex figure test: A test for the assessment of planning, visual spatial ability, and memory in older adults. *Appl. Neuropsychol. Adult*, 27(2):101–107, 2020.

[143] Katherine L Possin, Victor R Laluz, Oscar Z Alcantar, Bruce L Miller, and Joel H Kramer. Distinct neuroanatomical substrates and cognitive mechanisms of figure copy performance in alzheimer's disease and behavioral variant frontotemporal dementia. *Neuropsychologia*, 49(1):43–48, January 2011.

[144] Ronald B Postuma, Werner Poewe, Irene Litvan, Simon Lewis, Anthony E Lang, Glenda Halliday, Christopher G Goetz, Piu Chan, Elizabeth Slow, Klaus Seppi, Eva Schaffer, Silvia Rios-Romenets, Taomian Mi, Corina Maetzler, Yuan Li, Beatrice Heim, Ian O Bledsoe, and Daniela Berg. Validation of the MDS clinical diagnostic criteria for parkinson's disease. *Mov. Disord.*, 33(10):1601–1608, October 2018.

[145] Alexander Prange and Daniel Sonntag. Modeling users' cognitive performance using digital pen features. *Front Artif Intell*, 5:787179, May 2022.

[146] PyQT. PyQt reference guide. `http://www.riverbankcomputing.com/static/Docs/PyQt4/html/index.html`. Accessed: 2023-10-2.

[147] Anran Qi, Yulia Gryaditskaya, Tao Xiang, and Yi-Zhe Song. One sketch for all: One-shot personalized sketch segmentation. *arXiv [cs.CV]*, December 2021.

[148] Yonggang Qi and Zheng-Hua Tan. SketchSegNet+: An end-to-end learning of RNN for multi-class sketch semantic segmentation. *IEEE Access*, 7:102717–102726, 2019.

[149] Silvio Quincozes, Tubino Emilio, and Juliano Kazienko. MQTT protocol: Fundamentals, tools and future directions. *IEEE Lat. Am. Trans.*, 17(09):1439–1448, September 2019.

[150] J R Quinlan. Induction of decision trees. *Mach. Learn.*, 1(1):81–106, March 1986.

[151] A Rey. L'examen psychologique dans les cas d'encephalopathie traumatique. *Arch. Psychol.*, 28:286–340, 1941.

[152] André Rey and P A Osterrieth. Translations of excerpts from andre rey"s psychological examination of traumatic encephalopathy and P. a. osterrieth"s the complex figure copy test. *Clin. Neuropsychol.*, 7(1):4–21.

[153] Peter Riederer, Daniela Berg, Nicolas Casadei, Fubo Cheng, Joseph Classen, Christian Dresel, Wolfgang Jost, Rejko Krüger, Thomas Müller, Heinz Reichmann, Olaf Rieß, Alexander Storch, Sabrina Strobel, Thilo van Eimeren, Hans-Ullrich Völker, Jürgen

Winkler, Konstanze F Winklhofer, Ullrich Wüllner, Friederike Zunke, and Camelia-Maria Monoranu. -synuclein in parkinson's disease: causal or bystander? *J. Neural Transm.*, 126(7):815–840, July 2019.

[154] Giovanni Rizzo, Massimiliano Copetti, Simona Arcuti, Davide Martino, Andrea Fontana, and Giancarlo Logroscino. Accuracy of clinical diagnosis of parkinson disease: A systematic review and meta-analysis. *Neurology*, 86(6):566–576, February 2016.

[155] Daniel Rodríguez-Martín, Joan Cabestany, Carlos Pérez-López, Marti Pie, Joan Calvet, Albert Samà, Chiara Capra, Andreu Català, and Alejandro Rodríguez-Molinero. A new paradigm in parkinson's disease evaluation with wearable medical devices: A review of STAT-ONTM. *Front. Neurol.*, 13:912343, June 2022.

[156] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *arXiv [cs.CV]*, May 2015.

[157] F Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychol. Rev.*, 65(6):386–408, November 1958.

[158] Sara Rosenblum, Margalit Samuel, Sharon Zlotnik, Ilana Erikh, and Ilana Schlesinger. Handwriting as an objective tool for parkinson's disease diagnosis. *J. Neurol.*, 260(9):2357–2361, September 2013.

[159] G Webster Ross, Helen Petrovitch, Robert D Abbott, James Nelson, William Markesbery, Daron Davis, John Hardman, Lenore Launer, Kamal Masaki, Caroline M Tanner, and Lon R White. Parkinsonian signs and substantia nigra neuron density in decendents elders without PD. *Ann. Neurol.*, 56(4):532–539, October 2004.

[160] Alejandro Pasos Ruiz, Michael Flynn, James Large, Matthew Middlehurst, and Anthony Bagnall. The great multivariate time series classification bake off: a review and experimental evaluation of recent algorithmic advances. *Data Min. Knowl. Discov.*, 35(2):401–449, 2021.

[161] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, October 1986.

[162] Christoph Sager, Christian Janiesch, and Patrick Zschech. A survey of image labelling for computer vision applications. *Journal of Business Analytics*, 4(2):91–110, July 2021.

[163] Steven L Salzberg. C4.5: Programs for machine learning by J. ross quinlan. morgan kaufmann publishers, inc., 1993. *Mach. Learn.*, 16(3):235–240, September 1994.

[164] Yuko Sano, Akihiko Kandori, Keisuke Shima, Yuki Yamaguchi, Toshio Tsuji, Masafumi Noda, Fumiko Higashikawa, Masaru Yokoe, and Saburo Sakoda. Quantifying

parkinson's disease finger-tapping severity by extracting and synthesizing finger motion properties. *Med. Biol. Eng. Comput.*, 54(6):953–965, June 2016.

[165] Iqbal H Sarker. Deep learning: A comprehensive overview on techniques, taxonomy, applications and research directions. *SN Comput. Sci.*, 2(6):420, August 2021.

[166] Federica Scarpina, Erika Ambiel, Giovanni Albani, Luca Guglielmo Pradotto, and Alessandro Mauro. Utility of boston qualitative scoring system for rey-osterrieth complex figure: evidence from a parkinson's diseases sample. *Neurol. Sci.*, 37(10):1603–1611, October 2016.

[167] Luis F Schettino, Sergei V Adamovich, and Howard Poizner. Effects of object shape and visual feedback on hand configuration during grasping. *Exp. Brain Res.*, 151(2):158–166, July 2003.

[168] A Schrag, Y Ben-Shlomo, and N P Quinn. Cross sectional prevalence survey of idiopathic parkinson's disease and parkinsonism in london. *BMJ*, 321(7252):21–22, July 2000.

[169] C E Shannon. Communication in the presence of noise. *Proceedings of the IRE*, 37(1):10–21, January 1949.

[170] Hae-Won Shin and Sun Ju Chung. Drug-induced parkinsonism. *J. Clin. Neurol.*, 8(1):15–21, March 2012.

[171] Min-Sup Shin, Sun-Young Park, Se-Ran Park, Soon-Ho Seol, and Jun Soo Kwon. Clinical and empirical applications of the rey-osterrieth complex figure test. *Nat. Protoc.*, 1(2):892–899, 2006.

[172] Krista G Sibley, Christine Girges, Ehsan Hoque, and Thomas Foltynie. Video-based analyses of parkinson's disease severity: A brief review. *J. Parkinsons. Dis.*, 11(s1):S83–S93, 2021.

[173] Leslie N Smith. Cyclical learning rates for training neural networks. *arXiv [cs.CV]*, June 2015.

[174] Leslie N Smith. A disciplined approach to neural network hyper-parameters: Part 1 – learning rate, batch size, momentum, and weight decay. *arXiv [cs.LG]*, March 2018.

[175] Irwin Sobel. An isotropic 3x3 image gradient operator. February 2014.

[176] Jessica Somerville. *A Comparison of Administration Procedures for the Rey-Osterrieth Complex Figure: Flow-Charts vs. Pen-Switching.* PhD thesis, University of Rhode Island, 2000.

[177] Nitish Srivastava, Geoffrey E Hinton, A Krizhevsky, I Sutskever, and R Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15:1929–1958, 2014.

[178] Julien Stamatakis, Jérome Ambroise, Julien Crémers, Hoda Sharei, Valérie Delvaux, Benoit Macq, and Gaëtan Garraux. Finger tapping clinimetric score prediction in parkinson's disease using low-cost accelerometers. *Comput. Intell. Neurosci.*, 2013:717853, April 2013.

[179] Jeffrey M Stanton. Galton, pearson, and the peas: A brief history of linear regression for statistics instructors. *J. Stat. Educ.*, 9(3), January 2001.

[180] Robert A Stern, Elizabeth A Singer, Lisa M Duke, Naomi G Singer, Clare E Morey, Emily W Daughtrey, and Edith Kaplan. The boston qualitative scoring system for the rey-osterrieth complex figure: Description and interrater reliability. *Clin. Neuropsychol.*, 8(3):309–322, August 1994.

[181] Shota Suzumura, Yoshikiyo Kanada, Aiko Osawa, Junpei Sugioka, Natsumi Maeda, Taishi Nagahama, Kenta Shiramoto, Katsumi Kuno, Shiori Kizuka, Yuko Sano, Tomohiko Mizuguchi, Akihiko Kandori, and Izumi Kondo. Assessment of finger motor function that reflects the severity of cognitive function. *Fujita Med J*, 7(4):122–129, 2021.

[182] Chang Wei Tan, Angus Dempster, Christoph Bergmeir, and Geoffrey I Webb. MultiRocket: Multiple pooling operators and transformations for fast and effective time series classification. *arXiv [cs.LG]*, January 2021.

[183] Laughlin B Taylor. Localisation of cerebral lesions by psychological testing: Chapter XIV. *Neurosurgery*, 16:269, January 1969.

[184] H L Teulings, J L Contreras-Vidal, G E Stelmach, and C H Adler. Parkinsonism reduces coordination of fingers, wrist, and arm in fine motor control. *Exp. Neurol.*, 146(1):159–170, July 1997.

[185] Mathew Thomas, Abhishek Lenka, and Pramod Kumar Pal. Handwriting analysis in parkinson's disease: Current status and future directions. *Mov Disord Clin Pract*, 4(6):806–818, November 2017.

[186] Robert Tibshirani. Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Series B Stat. Methodol.*, 58(1):267–288, January 1996.

[187] Marianne Tiihonen, Britta U Westner, Markus Butz, and Sarang S Dalal. Parkinson's disease patients benefit from bicycling - a systematic review and meta-analysis. *NPJ Parkinsons Dis.*, 7(1):86, September 2021.

[188] Francisco J Valverde-Albacete and Carmen Peláez-Moreno. 100% classification accuracy considered harmful: the normalized information transfer factor explains the accuracy paradox. *PLoS One*, 9(1):e84217, January 2014.

[189] Guido Van Rossum and Fred L Drake. *Python 3 Reference Manual: (Python Documentation Manual Part 2).* CreateSpace Independent Publishing Platform, March 2009.

[190] J R Vane and Others. Adrenergic mechanisms:(ciba foundation symposium jointly with committee for symposia on drug action), 1960.

[191] Aleš Vysocký, Stefan Grushko, Petr Oščádal, Tomáš Kot, Ján Babjak, Rudolf Jánoš, Marek Sukop, and Zdenko Bobovský. Analysis of precision and stability of hand tracking with leap motion sensor. *Sensors*, 20(15), July 2020.

[192] Geoff Walker. A review of technologies for sensing contact location on the surface of a display. *J. Soc. Inf. Disp.*, 20(8):413–440, August 2012.

[193] Sam S Webb, Margaret Jane Moore, Anna Yamshchikova, Valeska Kozik, Mihaela D Duta, Irina Voiculescu, and Nele Demeyere. Validation of an automated scoring program for a digital complex figure copy task within healthy aging and stroke. *Neuropsychology*, 35(8):847–862, November 2021.

[194] Myron F Weiner, Linda S Hynan, Heidi Rossetti, and Jed Falkowski. Luria's three-step test: what is it and what does it tell us? *Int. Psychogeriatr.*, 23(10):1602–1606, December 2011.

[195] Bernard Widrow and Marcian E Hoff. Associative storage and retrieval of digital information in networks of adaptive "neurons". In *Biological Prototypes and Synthetic Systems*, pages 160–160. Springer US, Boston, MA, 1962.

[196] Bernard Widrow and Marcian E Hoff. (1960) bernard widrow and marcian E. hoff, "adaptive switching circuits," 1960 IRE WESCON convention record, new york: IRE, pp. 96-104. In *Neurocomputing, Volume 1*, pages 126–134. The MIT Press, April 1988.

[197] Stefan Williams, David Wong, Jane E Alty, and Samuel D Relton. Parkinsonian hand or clinician's eye? finger tap bradykinesia interrater reliability for 21 movement disorder experts. *J. Parkinsons. Dis.*, 13(4):525–536, 2023.

[198] Lumin Yang, Jiajie Zhuang, Hongbo Fu, Xiangzhi Wei, Kun Zhou, and Youyi Zheng. SketchGNN: Semantic sketch segmentation with graph neural networks. *arXiv [cs.CV]*, March 2020.

[199] Zhao Yin, Victor J Geraedts, Ziqi Wang, Maria Fiorella Contarino, Hamdi Dibeklioglu, and Jan van Gemert. Assessment of parkinson's disease severity from videos using deep architectures. *IEEE J. Biomed. Health Inform.*, 26(3):1164–1176, March 2022.

[200] M Yokoe, R Okuno, T Hamasaki, Y Kurachi, K Akazawa, and S Sakoda. Opening velocity, a novel parameter, for finger tapping test in patients with parkinson's disease. *Parkinsonism Relat. Disord.*, 15(6):440–444, July 2009.

[201] Nan-Ying Yu, Arend W A Van Gemmert, and Shao-Hsia Chang. Characterization of graphomotor functions in individuals with parkinson's disease and essential tremor. *Behav. Res. Methods*, 49(3):913–922, June 2017.

[202] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. CutMix: Regularization strategy to train strong classifiers with localizable features. *arXiv [cs.CV]*, May 2019.

[203] Poonam Zham, Dinesh K Kumar, Peter Dabnichki, Sridhar Poosapadi Arjunan, and Sanjay Raghav. Distinguishing different stages of parkinson's disease using composite index of speed and pen-pressure of sketching a spiral. *Front. Neurol.*, 8:435, September 2017.

[204] Aston Zhang, Zachary C Lipton, Mu Li, and Alexander J Smola. Dive into deep learning. *arXiv [cs.LG]*, June 2021.

[205] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv [cs.LG]*, October 2017.

[206] Zhen-Xin Zhang, Zhen-Hua Dong, and Gustavo C Román. Early descriptions of parkinson disease in ancient china. *Arch. Neurol.*, 63(5):782–784, May 2006.

[207] Aite Zhao and Jianbo Li. Two-channel lstm for severity rating of parkinson's disease using 3d trajectory of hand motion. *Multimed. Tools Appl.*, 81(23):33851–33866, September 2022.

[208] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. *Proc. Conf. AAAI Artif. Intell.*, 34(07):13001–13008, April 2020.